



Hadoop MapReduce

• • •

CS157B Frank Mock Spring 2017

Some things I Will Discuss

1. Installing Big Data platform - HortonWorks
2. Yelp Data Ingestion Process
3. Briefly Overview of MapReduce
4. How to design MapReduce job
5. Steps to run HelloWorld example (a.k.a. WordCount)
6. Hadoop data types
7. My MapReduce job implementation
8. MapReduce Job Optimization
9. What I learned

Big Data Platform



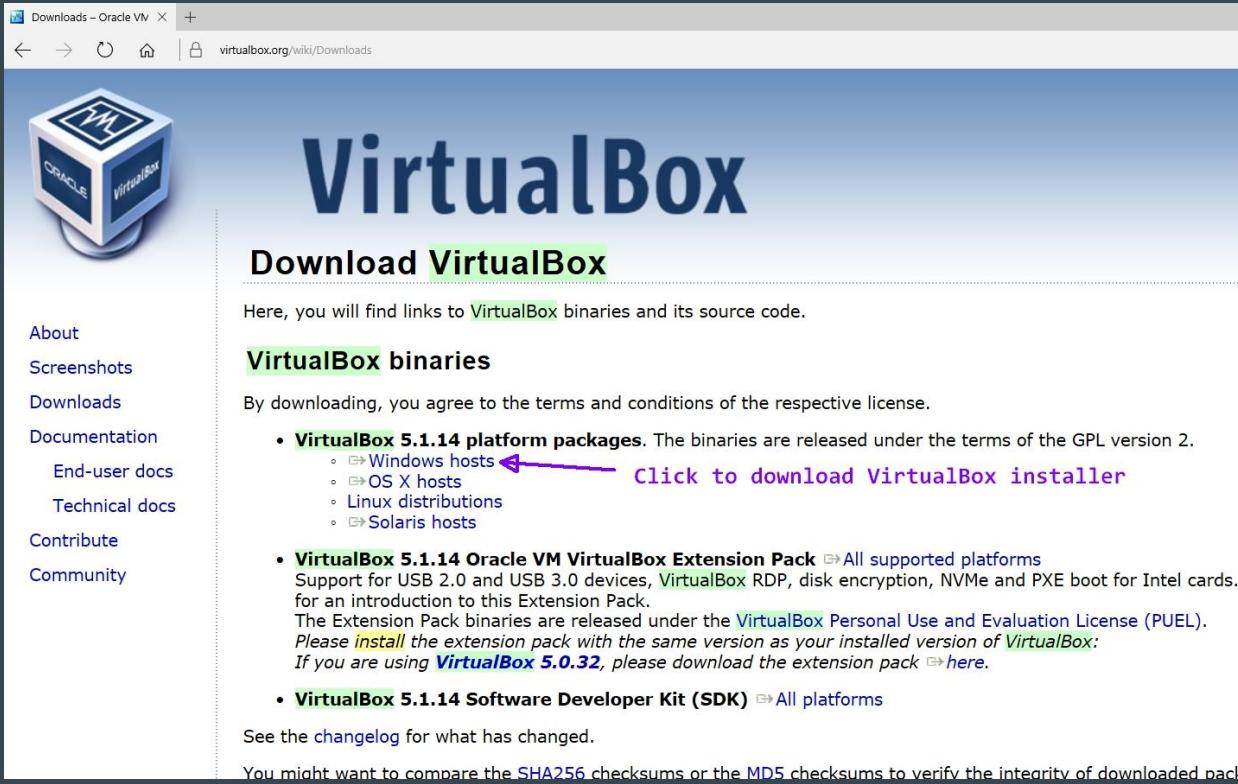
HortonWorks is an all-in-one Hadoop Big Data platform

HortonWorks includes Hadoop MapReduce

And it includes other higher-level Big Data tools that abstract away the gorey stuff

Like Hive, Pig, etc.

Must Install a Virtual Machine to Use HortonWorks



A screenshot of a web browser window titled "Downloads – Oracle VM". The address bar shows "virtualbox.org/wiki/Downloads". The main content is the VirtualBox download page. On the left is a sidebar with links: About, Screenshots, Downloads, Documentation, End-user docs, Technical docs, Contribute, and Community. The main area features a large blue "VirtualBox" logo and a "Download VirtualBox" button. Below it, text says "Here, you will find links to VirtualBox binaries and its source code." A section titled "VirtualBox binaries" lists download options: "VirtualBox 5.1.14 platform packages", "VirtualBox 5.1.14 Oracle VM VirtualBox Extension Pack", and "VirtualBox 5.1.14 Software Developer Kit (SDK)". A purple arrow points from the text "Click to download VirtualBox installer" to the "Windows hosts" link under the first bullet point. The "Windows hosts" link is highlighted with a purple box.

VirtualBox

Download VirtualBox

Here, you will find links to VirtualBox binaries and its source code.

VirtualBox binaries

By downloading, you agree to the terms and conditions of the respective license.

- **VirtualBox 5.1.14 platform packages.** The binaries are released under the terms of the GPL version 2.
 - Windows hosts
 - OS X hosts
 - Linux distributions
 - Solaris hosts
- **VirtualBox 5.1.14 Oracle VM VirtualBox Extension Pack** ◦ All supported platforms
Support for USB 2.0 and USB 3.0 devices, VirtualBox RDP, disk encryption, NVMe and PXE boot for Intel cards. for an introduction to this Extension Pack.
The Extension Pack binaries are released under the [VirtualBox Personal Use and Evaluation License \(PUEL\)](#).
Please install the extension pack with the same version as your installed version of VirtualBox:
If you are using [VirtualBox 5.0.32](#), please download the extension pack ◦ here.
- **VirtualBox 5.1.14 Software Developer Kit (SDK)** ◦ All platforms

See the [changelog](#) for what has changed.

You might want to compare the [SHA256](#) checksums or the [MD5](#) checksums to verify the integrity of downloaded pack

Download Version of HortonWorks for Your VM

stallingHortonworksSandbox | New tab Hortonworks Connector +

hortonworks.com/downloads/#sandbox

SANDBOX DATAFLOW DATA PLATFORM TECH PREVIEW

Hortonworks Sandbox on a VM

HDP® 2.5 on Hortonworks Sandbox

Tutorials		Release Notes		Import on Virtual Box		MD5 : d42a9bd11f29775cc5b804ce82a72efd	DOWNLOAD FOR VIRTUALBOX
Tutorials		Release Notes		Import on VMware		MD5 : f1d45e93ab9f2a655db559be5b2f2f43	DOWNLOAD FOR VMWARE
Tutorials		Release Notes		Import on Docker		MD5 : c613fab7ed21e15886ab23d7a28aec8a	DOWNLOAD FOR DOCKER

I downloaded this since I'm using VirtualBox

DOWNLOADED

Hortonworks Sandbox in the Cloud

HDP 2.4 on Azure with Hortonworks Sandbox

Tutorial: Sandbox on Azure		Try it one month for free	ONE MONTH TRIAL
----------------------------	--	---------------------------	------------------------

Followed Instructions Supplied by HortonWorks

<http://hortonworks.com/wp-content/uploads/unversioned/pdfs/InstallingHortonworksSandbox2onWindowsusingVB.pdf>

Installing Hortonworks Sandbox 2.0 – VirtualBox on Windows

Getting Ready to install on Windows using Oracle VirtualBox

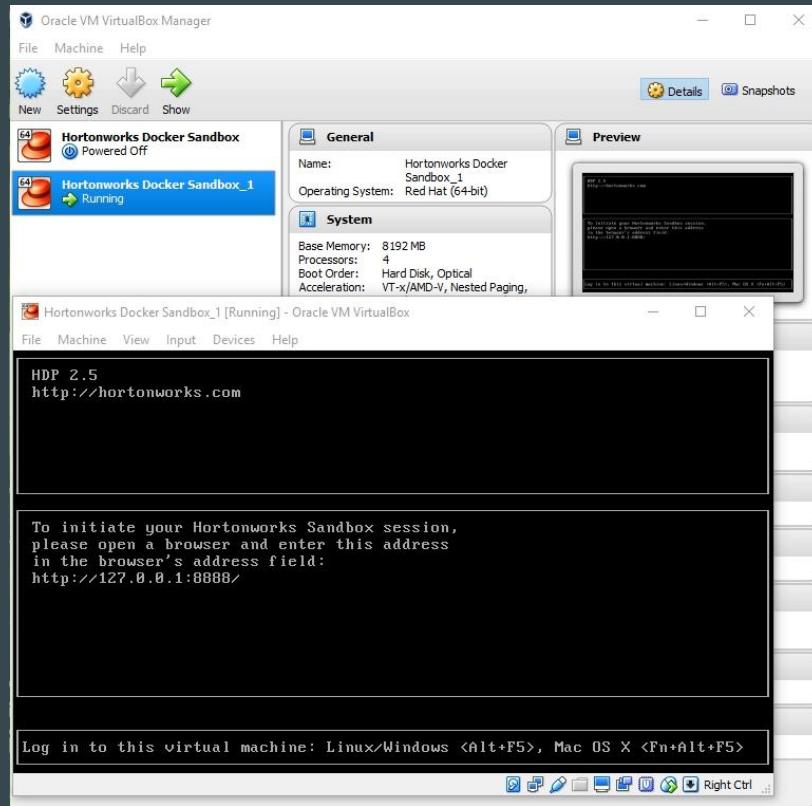
Use this section to prepare for your installation.

Prerequisites

To use the Hortonworks Sandbox on Windows you must have the following resources available to you:

- Hosts:
I had to enable virtualization in my computer's BIOS
 - A 64-bit machine with a chip that supports virtualization. Not all 64-bit chips have this capability. Check your system documentation or your IT department. For more information, see this Microsoft article: <http://windows.microsoft.com/en-us/windows7/32-bit-and-64-bit-windows-frequently-asked-questions>
 - A BIOS that has been set to enable virtualization support. This is usually already set, but in some cases must be set manually.

Running HortonWorks



After installation, start the HortonWorks virtual machine in VirtualBox.

It takes a minute to startup - be patient

After startup, open a web browser and go to <http://127.0.0.1:8888/> to login

Be sure to disable pop-up blocking in your web browser

After entering URL in Web Browser

The screenshot shows a web browser window with the title "Hortonworks Sandbox W" and the address bar displaying "localhost:8888". The page content is the Hortonworks Sandbox landing page. At the top center is the Hortonworks logo with three elephants and the text "HORTONWORKS POWERING THE FUTURE OF DATA™". Below the logo is a large orange hexagon containing the word "SANDBOX". To the right of the hexagon is the text "HDP2.5". On the left side, there is a green hexagon containing the "HDP" logo with "Hortonworks Data Platform" and "powered by Apache Hadoop" text. Below this logo is the heading "NEW TO HDP" and the subtext "Explore the Hortonworks Data Platform (HDP) Walk through a typical use case with the tutorial". A purple arrow points from the text "Click to launch sign-in page" to a green button labeled "LAUNCH DASHBOARD". On the right side, there is a blue hexagon containing three gears, with the heading "ADVANCED HDP" and the subtext "Expand your Hortonworks Data Platform (H Access components in Sandbo". A green button labeled "QUICK LINKS" is located at the bottom right.

Hortonworks Sandbox W X

localhost:8888

HORTONWORKS
POWERING THE FUTURE OF DATA™

SAND **BOX** HDP2.5

HDP
HORTONWORKS
DATA PLATFORM
powered by Apache Hadoop™

NEW TO HDP

Explore the Hortonworks Data Platform (HDP)
Walk through a typical use case with the tutorial

Click to launch sign-in page

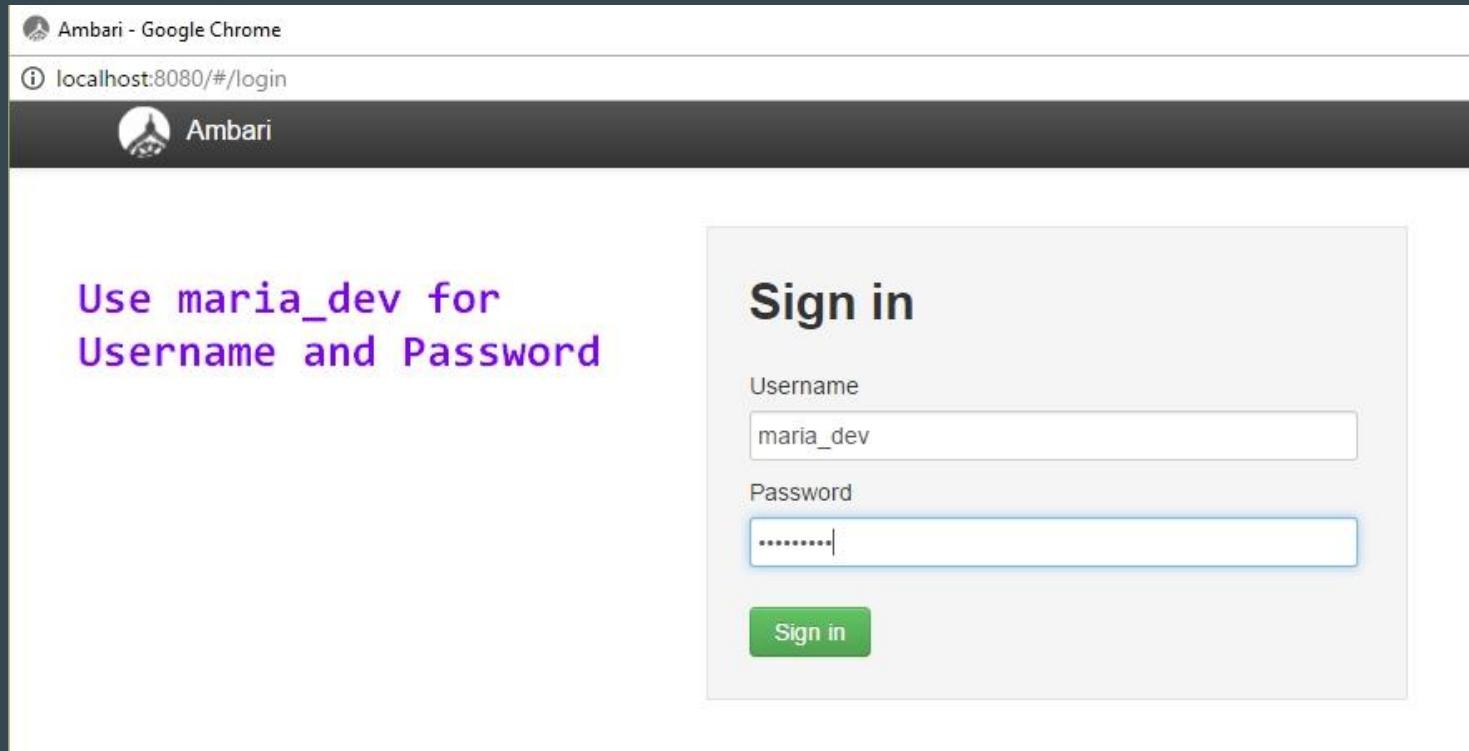
LAUNCH DASHBOARD

ADVANCED HDP

Expand your Hortonworks Data Platform (H
Access components in Sandbo

QUICK LINKS

Logging in to HortonWorks Sandbox



A screenshot of a web browser window showing the Ambari login interface. The title bar says "Ambari - Google Chrome" and the address bar shows "localhost:8080/#/login". The Ambari logo is in the top left corner of the main content area. On the left side of the content area, there is a purple text overlay that reads "Use maria_dev for Username and Password". The main content is a "Sign in" form with two input fields: "Username" containing "maria_dev" and "Password" containing a series of dots ("....."). A green "Sign in" button is at the bottom of the form.

Ambari - Google Chrome

localhost:8080/#/login

Ambari

Use maria_dev for
Username and Password

Sign in

Username

maria_dev

Password

.....

Sign in

Sandbox Dashboard

Ambari - Sandbox - Google Chrome
localhost:8080/#/main/dashboard/metrics

Ambari Sandbox 0 ops 0 alerts Dashboard Services Hosts Alerts Admin maria_dev ▾

Metrics Heatmaps Config History Metric Actions ▾ Last 1 hour ▾

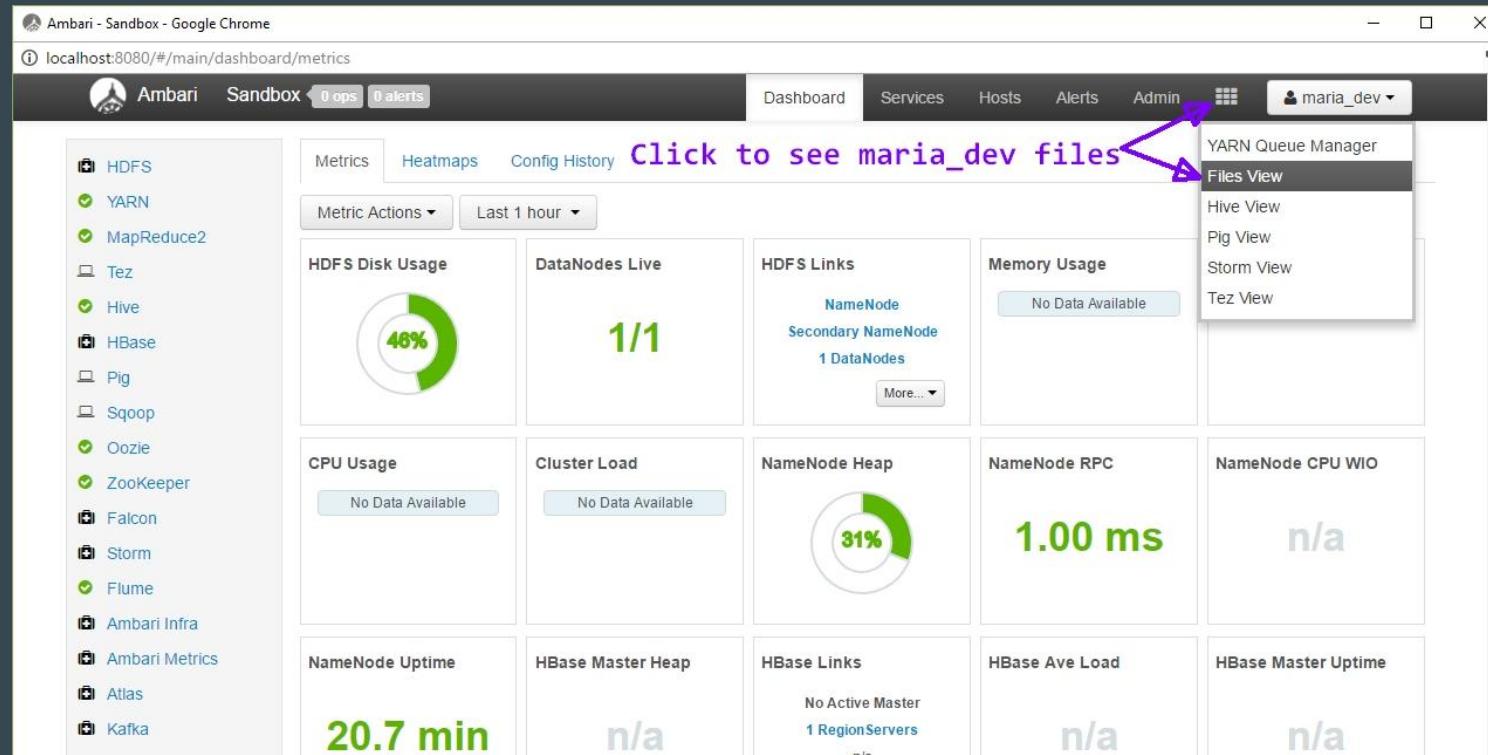
Click to see maria_dev files

HDFS Disk Usage: 48% DataNodes Live: 1/1 HDFS Links: NameNode, Secondary NameNode, 1 DataNodes Memory Usage: No Data Available

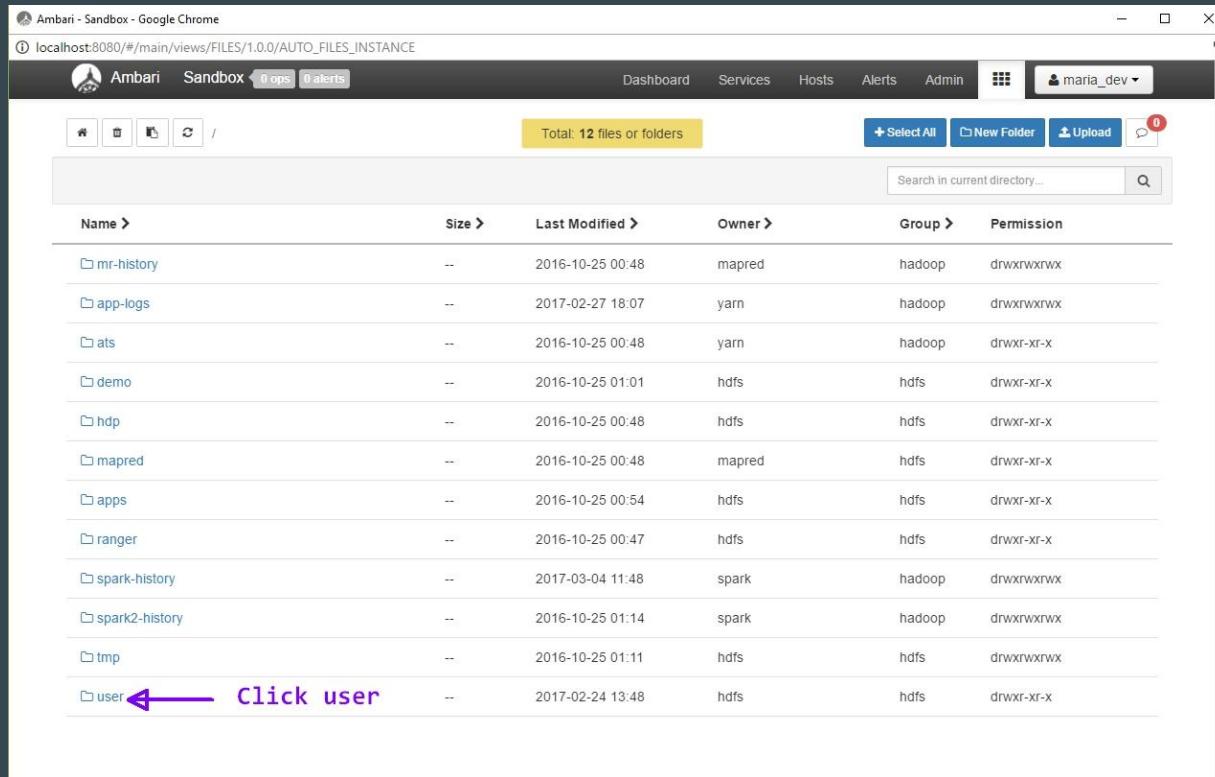
CPU Usage: No Data Available Cluster Load: No Data Available NameNode Heap: 31% NameNode RPC: 1.00 ms NameNode CPU WIO: n/a

NameNode Uptime: 20.7 min HBase Master Heap: n/a HBase Links: No Active Master, 1 RegionServers, n/a HBase Ave Load: n/a HBase Master Uptime: n/a

YARN Queue Manager
Files View
Hive View
Pig View
Storm View
Tez View



Navigating the File System



The screenshot shows the Ambari Sandbox interface with the URL `localhost:8080/#/main/views/FILES/1.0.0/AUTO_FILES_INSTANCE`. The top navigation bar includes links for Dashboard, Services, Hosts, Alerts, Admin, and a user dropdown for `maria_dev`. The main content area displays a file listing with the following details:

Name	Size	Last Modified	Owner	Group	Permission
mr-history	--	2016-10-25 00:48	mapred	hadoop	drwxrwxrwx
app-logs	--	2017-02-27 18:07	yarn	hadoop	drwxrwxrwx
ats	--	2016-10-25 00:48	yarn	hadoop	drwxr-xr-x
demo	--	2016-10-25 01:01	hdfs	hdfs	drwxr-xr-x
hdp	--	2016-10-25 00:48	hdfs	hdfs	drwxr-xr-x
mapred	--	2016-10-25 00:48	mapred	hdfs	drwxr-xr-x
apps	--	2016-10-25 00:54	hdfs	hdfs	drwxr-xr-x
ranger	--	2016-10-25 00:47	hdfs	hdfs	drwxr-xr-x
spark-history	--	2017-03-04 11:48	spark	hadoop	drwxrwxrwx
spark2-history	--	2016-10-25 01:14	spark	hadoop	drwxrwxrwx
tmp	--	2016-10-25 01:11	hdfs	hdfs	drwxrwxrwx
user	--	2017-02-24 13:48	hdfs	hdfs	drwxr-xr-x

A purple arrow points to the "user" entry in the list, with the text "Click user" written next to it.

Click user and on the next page click `maria_dev`

maria_dev Directories and Files

The screenshot shows the Ambari Sandbox interface for viewing files and folders. The URL is `localhost:8080/#/main/views/FILES/1.0.0/AUTO_FILES_INSTANCE`. The top navigation bar includes links for Dashboard, Services, Hosts, Alerts, Admin, and a user dropdown for maria_dev. Below the navigation is a toolbar with icons for Home, Delete, Copy, Paste, Refresh, and a search bar. The main area displays a list of 11 files and folders under the path `/ > user > maria_dev`. The list includes:

Name	Size	Last Modified	Owner	Group	Permission
test2	--	2017-02-27 22:37	maria_dev	hdfs	drwxr-xr-x
.Trash	--	2017-02-25 22:00	maria_dev	hdfs	drwx-----
business.csv	14.8 MB	2017-02-28 02:28	maria_dev	hdfs	-rw-r--r--
shakespeare.txt	150.0 kB	2017-02-27 18:00	maria_dev	hdfs	-rw-r--r--
test	--	2017-02-27 18:07	maria_dev	hdfs	drwxr-xr-x
.staging	--	2017-02-28 00:08	maria_dev	hdfs	drwx-----
test3	--	2017-02-27 23:40	maria_dev	hdfs	drwxr-xr-x
test4	--	2017-02-27 23:55	maria_dev	hdfs	drwxr-xr-x
test5	--	2017-02-28 00:03	maria_dev	hdfs	drwxr-xr-x
test6	--	2017-02-28 00:08	maria_dev	hdfs	drwxr-xr-x
testdata	--	2017-02-25 16:26	maria_dev	hdfs	drwxr-xr-x

Created Files From Yelp MySQL Data

I decided to re-create a subset of the Yelp data since it was ‘clean data’

Saved data as CSV files

```
mysql> (SELECT 'user_id', 'name', 'review_count', 'yelping_since', 'useful', 'funny', 'cool', 'fans', 'average_stars', 'compliment_hot',  
compliment_funny', 'compliment_writer', 'compliment_photos', 'type')  
-> Union  
-> (SELECT user_id, name, review_count, yelping_since, useful, funny, cool, fans, average_stars, compliment_hot, compliment_funny,  
, compliment_photos, type  
-> FROM user  
-> LIMIT 100000  
-> Into Outfile 'C:/ProgramData/MySQL/MySQL Server 5.6/Uploads/user.csv'  
-> Fields Enclosed By '\"' Terminated By ',' Escaped By '\"'  
-> Lines Terminated By '\r\n');  
Query OK, 100001 rows affected (1 min 36.65 sec)  
  
mysql> _
```

Getting Yelp data into HortonWorks

- Loaded a subset of the Yelp data - 100,000 records from each table
- Opted to use the command-line instead of GUI for data ingestion
- Learned that transferring a single file to Sandbox is **3-step process**
 1. `scp -P 2222 <local path of file> < user: path on remote HDFS>`
 2. `ssh maria_dev@127.0.0.1 -p 2222` to login to Hadoop HDFS
 3. `hdfs dfs -put <file_name> <dest. path in SandBox>`

Command-Line File Transfer a 3-Step Process

```
maria_dev@sandbox:~/testdata
General14@LAPTOP-5A28KKS4 MINGW64 ~
$ scp -P 2222 /C/Users/General14/Documents/School/CS157B/Project/testdata/review.csv maria_dev@127.0.0.1:/home/maria_de
v/testdata
maria_dev@127.0.0.1's password:
review.csv                                         100%   75MB  48.3MB/s  00:01
1

General14@LAPTOP-5A28KKS4 MINGW64 ~
$ ssh maria_dev@127.0.0.1 -p 2222
maria_dev@127.0.0.1's password:
Last Login: Sat Feb 25 20:59:36 2017 from 10.0.2.2
[maria_dev@sandbox ~]$ ls
testdata  testing  user3.csv
[maria_dev@sandbox ~]$ cd testdata
[maria_dev@sandbox testdata]$ ls
business_attributes.csv  business.csv  review.csv
3 [maria_dev@sandbox testdata]$ hdfs dfs -put review.csv /user/maria_dev/testdata/review.csv
[maria_dev@sandbox testdata]$
```

Yelp files in Sandbox

The screenshot shows the Ambari Sandbox interface. At the top, there's a header bar with the Ambari logo, 'Ambari - Sandbox - Google Chrome', and navigation links for Dashboard, Services, Hosts, Alerts, Admin. A user dropdown shows 'maria_dev'. Below the header is a toolbar with icons for file operations like New File, Delete, Copy, Paste, and a search bar. The main area displays a file listing for the directory '/ > user > maria_dev > testdata'. It shows five files: 'business.csv', 'business_attributes.csv', 'review.csv', 'tip.csv', and 'user.csv'. Each file entry includes its name, size (e.g., 14.8 MB, 5.1 MB), last modified date (e.g., 2017-02-25 12:45, 2017-02-25 13:00), owner ('maria_dev'), group ('hdfs'), and permission ('-rw-r--r--'). A yellow box highlights the message 'Total: 5 files or folders'.

Name >	Size >	Last Modified >	Owner >	Group >	Permission
business.csv	14.8 MB	2017-02-25 12:45	maria_dev	hdfs	-rw-r--r--
business_attributes.csv	5.1 MB	2017-02-25 13:00	maria_dev	hdfs	-rw-r--r--
review.csv	74.7 MB	2017-02-25 13:14	maria_dev	hdfs	-rw-r--r--
tip.csv	13.0 MB	2017-02-25 16:26	maria_dev	hdfs	-rw-r--r--
user.csv	11.9 MB	2017-02-25 16:17	maria_dev	hdfs	-rw-r--r--

What is MapReduce?



Programming model designed for ‘Big Data’ batch processing

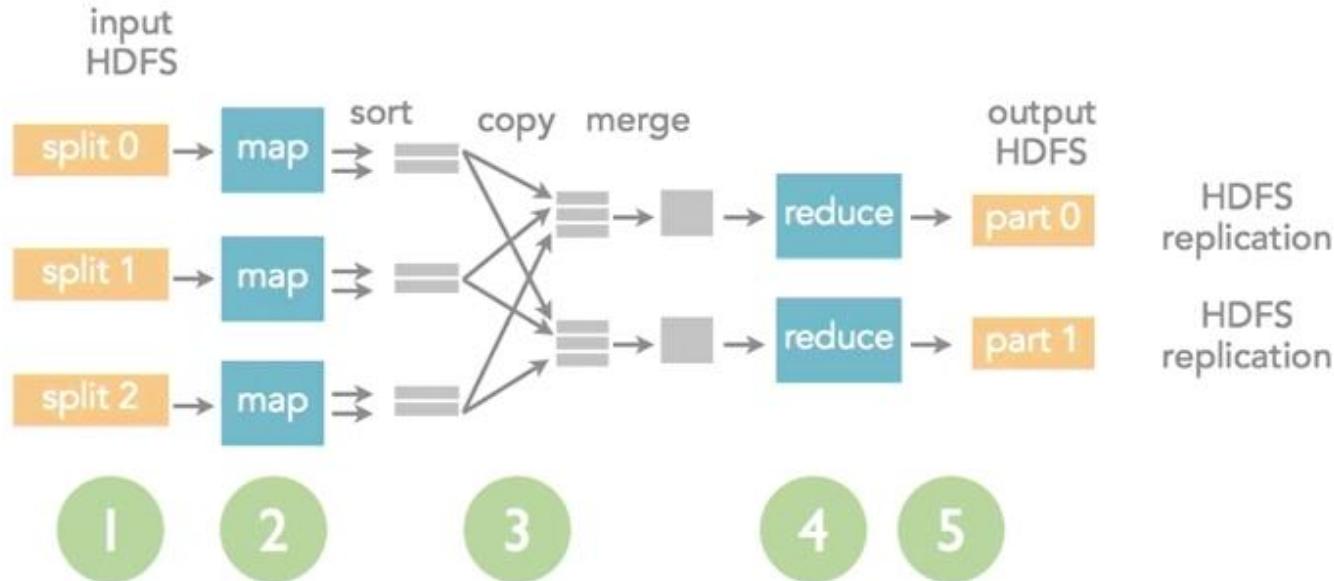
Originally created out of research by Google - (designed to index the internet)

There are two parts:

- Map
- Reduce

To use MapReduce to solve problem you must decompose your problem into a Map task and a Reduce task. Based on the problem, reduction may not be necessary.

Visualizing MapReduce



First Run Practice MapReduce Job

Did the HelloWorld MapReduce tutorial on HortonWorks

<https://hortonworks.com/hadoop-tutorial/introducing-apache-hadoop-developers/>

MapReduce Java source code can be copied from above URL

Must add lots of JAR files to project buildpath

Hadoop JAR files

```
import java.io.IOException;
import java.util.*;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.conf.*;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

public class WordCount2 {

    public static class Map extends 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, Context context) throws IOException,
            String line = value.toString();
            StringTokenizer tokenizer = new StringTokenizer(line);
            while (tokenizer.hasMoreTokens()) {
                word.set(tokenizer.nextToken());
                context.write(word, one);
            }
        }

        public static class Reduce extends Reducer<Text, IntWritable, Text, IntWritable> {
```

<http://mvnrepository.com/artifact/org.apache.hadoop>

Maven Repository: org.apache.hadoop

Indexed Artifacts (5.95M)

Popular Categories

- Aspect Oriented
- Actor Frameworks
- Application Metrics
- Build Tools
- Bytecode Libraries
- Command Line Parsers
- Cache Implementations
- Cloud Computing
- Code Analyzers
- Collections
- Configuration Libraries
- Core Utilities
- Date and Time Utilities
- Dependency Injection
- Embedded SQL Databases

Home » org.apache.hadoop

Group: org.apache.hadoop

- Apache Hadoop Common**
org.apache.hadoop » hadoop-common
Apache Hadoop Common
975 usages Apache
- Apache Hadoop Client Aggregator**
org.apache.hadoop » hadoop-client
Apache Hadoop Client aggregation pom with dependencies exposed
667 usages Apache
- Hadoop Core**
org.apache.hadoop » hadoop-core
Hadoop Core
664 usages Apache
- Apache Hadoop HDFS**
org.apache.hadoop » hadoop-hdfs
Apache Hadoop HDFS
535 usages Apache
- Apache Hadoop MapReduce Core**
org.apache.hadoop » hadoop-mapreduce-client-core
Apache Hadoop MapReduce Core
391 usages Apache



The Jar files can be found here

```
39  
40     public static void main(String[] args) throws Exception {  
41         Configuration conf = new Configuration();  
42  
43         Job job = new Job(conf, "WordCount2");  
44         job.setJarByClass(WordCount2.class);  
45  
46         job.setOutputKeyClass(Text.class);  
47         job.setOutputValueClass(IntWritable.class);  
48  
49         job.setMapperClass(Map.class);  
50     }
```

This line is missing from example code.
Hadoop finds relevant JAR by looking for the JAR file containing this class.

Problems @ Javadoc Declaration Console

<terminated> PostalCodeRating [Java Application] C:\Program Files (x86)\Java\jre1.8.0_111\bin\javaw.exe (Feb 28, 2017, 3:31:54 PM)

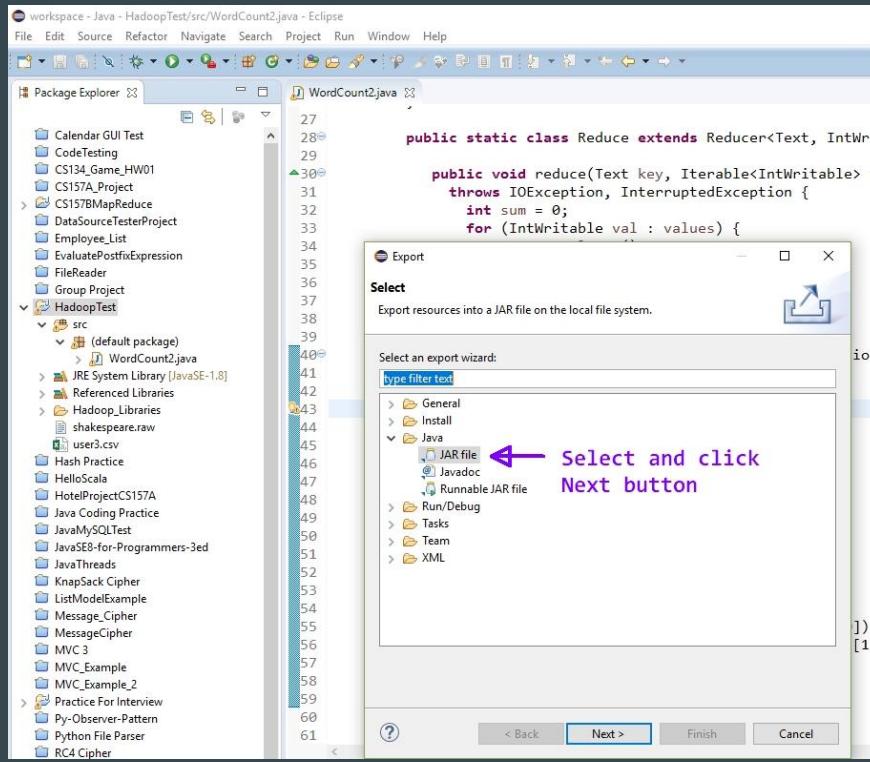
Exception in thread "main" java.lang.NoClassDefFoundError: org/apache/commons/logging/LogFactory
at org.apache.hadoop.conf.Configuration.<clinit>([Configuration.java:177](#))
at PostalCodeRating.main([PostalCodeRating.java:79](#))

Caused by: java.lang.ClassNotFoundException: org.apache.commons.logging.LogFactory
at java.net.URLClassLoader.findClass(Unknown Source)
at java.lang.ClassLoader.loadClass(Unknown Source)
at sun.misc.Launcher\$AppClassLoader.loadClass(Unknown Source)
at java.lang.ClassLoader.loadClass(Unknown Source)

... 2 more

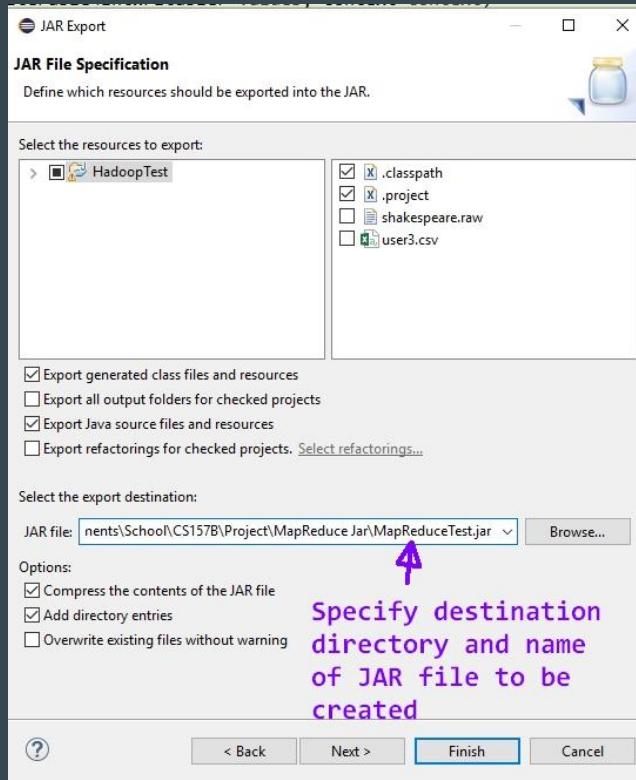
Doesn't run in Eclipse, but the JAR runs in Hadoop

How to make a JAR File in Eclipse



1. Right-Click the project name and select Export
2. Select Jar File and click Next

How to make a JAR File in Eclipse (continued)



3. Tell Eclipse where to create the JAR and give it a name.

4. Click Finish

5. Next, transfer the JAR to the HDFS

How to Run a JAR file (MapReduce Job) in Hadoop

Login to sandbox terminal and use the `jar` Hadoop command.

Usage: `hadoop jar <jar_file_name> [mainClass] args...`

```
hadoop jar MapReduceTest.jar WordCount2 shakespeare.txt /user/maria_dev/test
```

This directory should not already exist



Job will abort if it does. Prevents loss of previous data analysis

Transfer to HDFS and Run the Test MapReduce Job

```
MINGW32:~  
MapReduceTest.jar shakespeare.txt testdata testing user3.csv  
[maria_dev@sandbox ~]$ hdfs dfs -put shakespeare.txt /user/maria_dev/shakespeare.txt  
[maria_dev@sandbox ~]$ hadoop jar MapReduceTest.jar WordCount2 shakespeare.txt /user/maria_dev/test  
17/02/28 02:07:26 INFO impl.TimelineClientImpl: Timeline service address: http://sandbox.hortonworks.com:8188/ws/v1/timeline/  
17/02/28 02:07:26 INFO client.RMProxy: Connecting to ResourceManager at sandbox.hortonworks.com/172.17.0.2:8050  
17/02/28 02:07:26 INFO client.AHSProxy: Connecting to Application History server at sandbox.hortonworks.com/172.17.0.2:10200  
17/02/28 02:07:27 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with  
17/02/28 02:07:28 INFO input.FileInputFormat: Total input paths to process : 1  
17/02/28 02:07:28 INFO lzo.GPLNativeCodeLoader: Loaded native gpl library  
17/02/28 02:07:28 INFO lzo.LzoCodec: Successfully loaded & initialized native-lzo library [hadoop-lzo rev 7a4b57bedce694048432dd5bf5b90a6c8ccdba80]  
17/02/28 02:07:28 INFO mapreduce.JobSubmitter: number of splits:1  
17/02/28 02:07:28 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1488246516397_0001  
17/02/28 02:07:29 INFO impl.YarnClientImpl: Submitted application application_1488246516397_0001  
17/02/28 02:07:30 INFO mapreduce.Job: The url to track the job: http://sandbox.hortonworks.com:8088/proxy/application_1488246516397_0001/  
17/02/28 02:07:30 INFO mapreduce.Job: Running job: job_1488246516397_0001  
17/02/28 02:07:46 INFO mapreduce.Job: Job job_1488246516397_0001 running in uber mode : false  
17/02/28 02:07:46 INFO mapreduce.Job: map 0% reduce 0%  
17/02/28 02:07:51 INFO mapreduce.Job: map 100% reduce 0%  
17/02/28 02:07:59 INFO mapreduce.Job: map 100% reduce 100%
```

MapReduce output is in test directory

Ambari - Sandbox - Google Chrome
localhost:8080/#/main/views/FILES/1.0.0/AUTO_FILES_INSTANCE

Ambari Sandbox 0 ops 0 alerts Dashboard Services Hosts Alerts Admin maria_dev

Total: 5 files or folders + Select All New Folder Upload Search in current directory...

Name >	Size >	Last Modified >	Owner >	Group >	Permission
↳					
↳ .Trash	--	2017-02-25 22:00	maria_dev	hdfs	drwx-----
↳ .staging	--	2017-02-27 18:08	maria_dev	hdfs	drwx-----
↳ shakespeare.txt	150.0 kB	2017-02-27 18:00	maria_dev	hdfs	-rw-r--r--
↳ test	--	2017-02-27 18:07	maria_dev	hdfs	drwxr-xr-x
↳ testdata	--	2017-02-25 16:26	maria_dev	hdfs	drwxr-xr-x

Success only means the test finished

Ambari - Sandbox - Google Chrome

localhost:8080/#/main/views/FILES/1.0.0/AUTO_FILES_INSTANCE

Ambari Sandbox 0 ops 0 alerts

Dashboard Services Hosts Alerts Admin maria_dev

Total: 2 files or folders

+ Select All New Folder Upload

/ > user > maria_dev > test

Search in current directory...

Name >	Size >	Last Modified >	Owner >	Group >	Permission
□_SUCCESS	0.1 kB	2017-02-27 18:07	maria_dev	hdfs	-rw-r--r--
□ part-r-00000	58.1 kB	2017-02-27 18:07	maria_dev	hdfs	-rw-r--r--



Ambari

Sandbox

0 ops

0 alerts

Dashboard

Services

Hosts

Alerts

Admin



File Preview

/user/maria_dev/test/part-r-00000

Name



_SUCCESS

part-r-00000

"Where	1
'A	2
'An	1
'Ay	1
'Ay';	1
'Ay.'	4
'By	1
'For	1
'God	1
'Heart's	4
'Hold,	1
'I	2
'I'	1
'I';	2
'I'll	1
'I,'	1
'I.'	1
'It	1
'Juliet,'	1

+ Select All

New Folder



Cancel

Download

Now implement my own MapReduce job

Coding a MapReduce Job

Problem: Find average review rating for all businesses in a specific zip code

Written as a SQL statement

```
SELECT postal_code, AVG(stars), count(*) AS business_count  
FROM business  
GROUP BY postal_code  
ORDER BY business_count DESC  
LIMIT 100;
```

MySQL Results

```
mysql> SELECT postal_code, AVG(stars), count(*) AS business_count
-> FROM business
-> GROUP BY postal_code
-> ORDER BY business_count DESC
-> LIMIT 100;
```

postal_code	AVG(stars)	business_count
89109	3.5757068452380953	2688
85251	3.9410935738444195	1774
85281	3.67276814386641	1557
89119	3.489802855200544	1471
...		
85253	3.9275568181818183	352
85248	3.690201729106628	347
85051	3.3771676300578033	346
89183	3.6246376811594203	345

100 rows in set (18.30 sec)

Using MapReduce

First break problem into two tasks:

- Map Task
- Reduce Task

Start by observing the data

I wrote a small Java program to see what I'm working with

```
4  
5 // MapReduce Job - Find average review rating for all businesses in a specific zip code  
6 public class AvgRatingBusinessZipcode {  
7     public static final String BUSINESS_CSV_LOCATION =  
8         "C:\\Users\\General4\\workspace\\YelpMapReduce\\business.csv";  
9  
10    private static void readInFile(String location) {  
11        try {  
12            BufferedReader reader = new BufferedReader(new FileReader(location));  
13            Scanner in = new Scanner(reader);  
14            String line = "";  
15            int lineCount = 0;  
16            while(in.hasNextLine() && lineCount++ < 10) {  
17                System.out.println(in.nextLine());  
18            }  
19            System.out.println();  
20            in.close();  
21        }  
22        catch(Exception ex) {  
23            ex.printStackTrace();  
}
```

Must extract data from this mess. Fun, right?



Problems @ Javadoc Declaration Console

terminated> AvgRatingBusinessZipcode [Java Application] C:\Program Files (x86)\Java\jre1.8.0_111\bin\javaw.exe (Feb 28, 2017, 10:44:40 AM)

```
business_id", "name", "neighborhood", "address", "city", "state", "postal_code", "latitude", "longitude", "stars", "review_count", "is_open", "type"  
--6MefnULPED_I942VcFNA", "John's Chinese BBQ Restaurant", "", "328 Highway 7 E, Chalmers Gate 11, Unit 10", "Richmond Hill", "ON", "L4B 3P7", "43.840905", "-79.39  
--7zmmkVg-IMGaXbuVd0SQ", "Primal Brewery", "", "16432 Old Statesville Rd", "Huntersville", "NC", "28078", "35.437086", "-80.843688", "4", "32", "1", "business"  
--9e10NYQuAa-CB_Rrw7Tw", "Delmonico Steakhouse", "The Strip", "3355 Las Vegas Blvd S", "Las Vegas", "NV", "89109", "36.123183", "-115.16919", "4", "1311", "1", "busin  
--9QQLMTbFzLJ_oT-ON3Xw", "Great Clips", "", "1835 E Guadalupe Rd, Ste 106", "Tempe", "AZ", "85283", "33.3616642", "-111.9096233", "3", "8", "1", "business"  
--ab39IjZR_xUf81WyTgH", "Famous Footwear", "", "1800 E Rio Salado Pky 110, Tempe Marketplace", "Tempe", "AZ", "85281", "33.4300928", "-111.9049649", "4", "9", "1", "  
--cgVkbWTig30YTkymKqA", "Eazor's Auto Salon", "", "616 Long Rd", "Pittsburgh", "PA", "15235", "40.45314", "-79.83886", "5", "6", "1", "business"  
--cjEBxM12obtaRHNSFrA", "Howl at the Moon", "Downtown", "125 7th St", "Pittsburgh", "PA", "15222", "40.443888", "-80.000223", "3", "39", "1", "business"  
--cZ6Hhc9F7vKXxHMVZSQ", "Pio Pio", "Dilworth", "1408 E Blvd", "Charlotte", "NC", "28203", "35.1998528426", "-80.8448199047", "4", "259", "1", "business"  
--DaPTJW3-tB1vP-PfdTEg", "Sunnyside Grill", "Corso Italia", "1218 Saint Clair Avenue W", "Toronto", "ON", "M6E", "43.6778069", "-79.4446742", "3.5", "32", "1", "busin
```

Zero in on target data

```
18     while(in.hasNextLine() && lineCount++ < 10) {  
19         String aLine = in.nextLine();  
20         System.out.println(aLine);  
21         aLine = aLine.replace("\\"", "");  
22         String[] colValue = aLine.split(",");  
23  
24         for(String n : colValue){  
25             System.out.println(n);  
26         }  
<
```

Problems @ Javadoc Declaration Console

<terminated> AvgRatingBusinessZipcode [Java Application] C:\Program Files (x86)\Java\jre1.8.0_111\bin\javaw.exe (Feb 28, 2017, 4:02:07 PM)

--cjBEbXMI2obtaRHNSFrA,"Howl at the Moon","Downtown","125 7th St","Pittsburgh","PA","15222","40.443888","-80.000223","3","39","1","business"

--cjBEbXMI2obtaRHNSFrA

Howl at the Moon

Downtown

125 7th St

Pittsburgh

PA

15222 ← Index 6

40.443888

-80.000223

3 ← Index 9

39

1

business

--cZ6Hhc9F7VkKXxHMVZSQ,"Pio Pio","Dilworth","1408 E Blvd","Charlotte","NC","28203","35.1998528426","-80.8448199047","4","259","1","business"

--cZ6Hhc9F7VkKXxHMVZSQ

Pio Pio

Dilworth

1408 E Blvd

Charlotte

Target data always at these indexes

Next write code using Hadoop MapReduce framework

hadoop.apache.org

Apache > Hadoop >



Search with Apache Solr Search Last Published: 01/26/2017 10:33:46

Top Wiki

About

- Welcome
- What Is Apache Hadoop...
- Getting Started ...
- Download Hadoop
- Who Uses Hadoop?...
- News
- Releases
- Release Versioning
- Mailing Lists
- Issue Tracking
- Who We Are?
- Who Uses Hadoop?
- Buy Stuff
- Sponsorship
- Thanks
- Privacy Policy
- Bylaws
- Committer criteria
- License

Documentation

Related Projects

built with Apache Forrest

Welcome to Apache™ Hadoop®!

What Is Apache Hadoop?

The Apache™ Hadoop® project develops open-source software for reliable, scalable, distributed computing.

The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

The project includes these modules:

- **Hadoop Common:** The common utilities that support the other Hadoop modules.
- **Hadoop Distributed File System (HDFS™):** A distributed file system that provides high-throughput access to application data.
- **Hadoop YARN:** A framework for job scheduling and cluster resource management.
- **Hadoop MapReduce:** A YARN-based system for parallel processing of large data sets.

Other Hadoop-related projects at Apache include:

- **Ambari™:** A web-based tool for provisioning, managing, and monitoring Apache Hadoop clusters which includes support for Hadoop HDFS, Hadoop MapReduce, Hive, HCatalog, HBase, ZooKeeper, Oozie, Pig and Sqoop. Ambari also provides a dashboard for viewing cluster health such as heatmaps and ability to view MapReduce, Pig and Hive applications visually alongwith features to diagnose their performance characteristics in a user-friendly manner.
- **Avro™:** A data serialization system.
- **Cassandra™:** A scalable multi-master database with no single points of failure.

Documentation for Hadoop Data Types

Hadoop provides its own set of basic types that are optimized for network serialization

They can be found at: [org.apache.hadoop.io package](https://hadoop.apache.org/docs/current/api/org/apache/hadoop/io/package-summary.html)

The screenshot shows a web browser displaying the Apache Hadoop documentation. The URL in the address bar is hadoop.apache.org/docs/current/api/org/apache/hadoop/io/package-summary.html. The page has a blue header with navigation links: OVERVIEW, PACKAGE (which is highlighted in orange), CLASS, USE, TREE, DEPRECATED, INDEX, and HELP. Below the header, there are links for PREV PACKAGE, NEXT PACKAGE, FRAMES, NO FRAMES, and ALL CLASSES. The main content area has a dark background with white text. The title is "Package org.apache.hadoop.io". The description below it reads: "Generic i/o code for use when reading and writing data to the network, to databases, and to files." At the bottom left, there is a link "See: Description".

Hadoop Wrapper Data Types

Hadoop	→	Java	Necessary Import
Text	→	String	import org.apache.hadoop.io.Text;
IntWritable	→	Integer	import org.apache.hadoop.io.IntWritable;
LongWritable	→	Long	import org.apache.hadoop.io.LongWritable;
FloatWritable	→	Float	import org.apache.hadoop.io.FloatWritable;

Inputs and Outputs

	(KEY, VALUE)	(Input Type, Output Type)
Mapper Input	= (Line byte offset, Text Line)	(LongWritable, Text) default
Mapper Output	= (PostalCode, Stars)	(Text, FloatWritable)
Reducer Input	= (PostalCode, Stars)	(Text, FloatWritable)
Reducer Output	= (PostalCode, Avg(stars))	(Text, FloatWritable)

Visualize the function inputs and outputs before coding

Map input < 0, "--ab39IjZR_xUf81WyTyHg","Famous Footwear","","1800 E Rio Salado ... "3","4","1","business" >

...

Map input < 100, "--cgVkbWTiga3OYTkymKqA","Eazor's Auto Salon","","616 Long Rd", ... "4","58","1","business" >

Map output <94582, 3.5 >

Map output <94582, 2.0 >

...

Map output <94582, 3.0 >

Map output <94582, 4.5 >

These will be grouped and sorted before being sent to reducer function

Reducer input < 94582, [3.5, 2.0, ... , 3.0, 4.5] > ← Reducer will get an input like this for each unique postal code

Reducer output 94582, 3.25 ← Entry in output file for each distinct postal code

The Map Function

```
public static class MapperClass
    extends Mapper<Object, Text, Text, FloatWritable> {

    Text postalCode = new Text();
    FloatWritable stars = new FloatWritable();

    public void map(Object key, Text value, Context context)
        throws IOException, InterruptedException {

        String line = value.toString(); // Convert to Java String

        line = line.replaceAll("\"", ""); // Remove double quotes
        String[] colValue = line.split(","); // Each token separated by comma

        postalCode.set(colValue[6]); // assign the 6th element to postalcode
        try {
            stars.set(Float.valueOf(colValue[9])); // assign 9th element to stars
        }
        catch (NumberFormatException e) {
            e.printStackTrace();
        }
        // Write output key, value pair to the context object
        context.write(postalCode, stars);
    }
}
```

The Reducer

```
public static class ReducerClass
    extends Reducer<Text, FloatWritable, Text, FloatWritable> {

    FloatWritable postalCodeRating = new FloatWritable();

    public void reduce(Text key, Iterable<FloatWritable> values, Context context)
        throws IOException, InterruptedException {
        // Record sum and count of stars
        float sum = 0;
        float count = 0;
        // Sum all the stars that share the same postal code
        for (FloatWritable val : values) {
            sum += val.get();
            count++;
        }
        // Compute the average
        postalCodeRating.set(sum / count);
        // Write output pair
        context.write(key, postalCodeRating);
    }
}
```

The MapReduce Job Runner

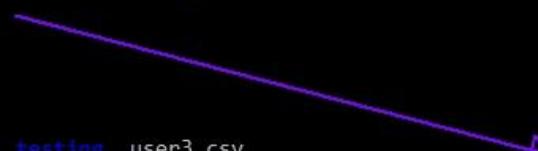
```
public static void main(String[] args) throws Exception {
    // Check for correct number of arguments
    if (args.length != 2) {
        System.out.println("usage: [input] [output]");
        System.exit(-1);
    }
    // Create and configure MapReduce job
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "PostalCodeRating");
    job.setJarByClass(PostalCodeRating.class);
    // Set Mapper and Reducer Class
    job.setMapperClass(MapperClass.class);
    job.setReducerClass(ReducerClass.class);
    // Set output types for keys and values
    job.setMapOutputKeyClass(Text.class);
    job.setMapOutputValueClass(FloatWritable.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(FloatWritable.class);
    // Set input and output format class
    job.setInputFormatClass(TextInputFormat.class); //default
    job.setOutputFormatClass(TextOutputFormat.class);
    // Set input and output files from arguments
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    // Send job to cluster and wait for job to complete
    job.waitForCompletion(true);
}
```

Run Job in Hadoop

MINGW32:~

```
PostalCodeRatingMapReduce.jar          100% 7521KB 7.3MB/s 00:01
General4@LAPTOP-5A28KKS4 ~
$ ssh maria_dev@127.0.0.1 -p 2222
maria_dev@127.0.0.1's password:
Last login: Tue Feb 28 08:02:41 2017 from 10.0.2.2
[maria_dev@sandbox ~]$ ls
MapReduceTest.jar PostalCodeRatingMapReduce.jar shakespeare.txt testdata testing user3.csv
[maria_dev@sandbox ~]$ hadoop jar PostalCodeRatingMapReduce.jar PostalCodeRating testdata/business.csv /user/maria_dev/test6
17/02/28 08:07:39 INFO impl.TimelineClientImpl: Timeline service address: http://sandbox.hortonworks.com:8188/ws/v1/timeline/
17/02/28 08:07:39 INFO client.RMProxy: Connecting to ResourceManager at sandbox.hortonworks.com/172.17.0.2:8050
17/02/28 08:07:39 INFO client.AHSProxy: Connecting to Application History server at sandbox.hortonworks.com/172.17.0.2:10200
17/02/28 08:07:40 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute
17/02/28 08:07:41 INFO input.FileInputFormat: Total input paths to process : 1
17/02/28 08:07:41 INFO lzo.GPLNativeCodeLoader: Loaded native gpl library
17/02/28 08:07:41 INFO lzo.LzoCodec: Successfully loaded & initialized native-lzo library [hadoop-lzo rev 7a4b57bedce694048432dd5bf5b90a6c8cc
17/02/28 08:07:41 INFO mapreduce.JobSubmitter: number of splits:1
17/02/28 08:07:41 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1488246516397_0006
17/02/28 08:07:42 INFO impl.YarnClientImpl: Submitted application application_1488246516397_0006
17/02/28 08:07:42 INFO mapreduce.Job: The url to track the job: http://sandbox.hortonworks.com:8088/proxy/application_1488246516397_0006/
17/02/28 08:07:42 INFO mapreduce.Job: Running job: job_1488246516397_0006
```

After several tests finally
was successful



```
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=15515127
HDFS: Number of bytes written=136326
HDFS: Number of read operations=6
HDFS: Number of large read operations=0
HDFS: Number of write operations=2

Job Counters
Launched map tasks=1
Launched reduce tasks=1
Data-local map tasks=1
Total time spent by all maps in occupied slots (ms)=4697
Total time spent by all reduces in occupied slots (ms)=4005
Total time spent by all map tasks (ms)=4697
Total time spent by all reduce tasks (ms)=4005
Total vcore-milliseconds taken by all map tasks=4697
Total vcore-milliseconds taken by all reduce tasks=4005
Total megabyte-milliseconds taken by all map tasks=1174250
Total megabyte-milliseconds taken by all reduce tasks=1001250

Map-Reduce Framework
Map input records=100001
Map output records=100001
Map output bytes=964985
Map output materialized bytes=1164993
Input split bytes=137
Combine input records=0
Combine output records=0
Reduce input groups=10560
Reduce shuffle bytes=1164993
Reduce input records=100001
Reduce output records=10560
Spilled Records=200002
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=422
CPU time spent (ms)=5110
Physical memory (bytes) snapshot=326160384
Virtual memory (bytes) snapshot=3891957760
Total committed heap usage (bytes)=154664960

Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
```

How long did the job take?

I'm not sure, about 10 seconds.

Time spent by all maps 4697 ms

Time spent by all reducers 4005 ms

Default configuration was used for this MapReduce job, which means only a single reducer is used.

Map output data size was less than HDFS block size so only one reduce task required

View the MapReduce Job Results

Ambari - Sandbox - Google Chrome

localhost:8080/#/main/views/FILES/1.0.0/AUTO_FILES_INSTANCE

Ambari Sandbox 0 ops 0 alerts Dashboard Services Hosts Alerts Admin maria_dev

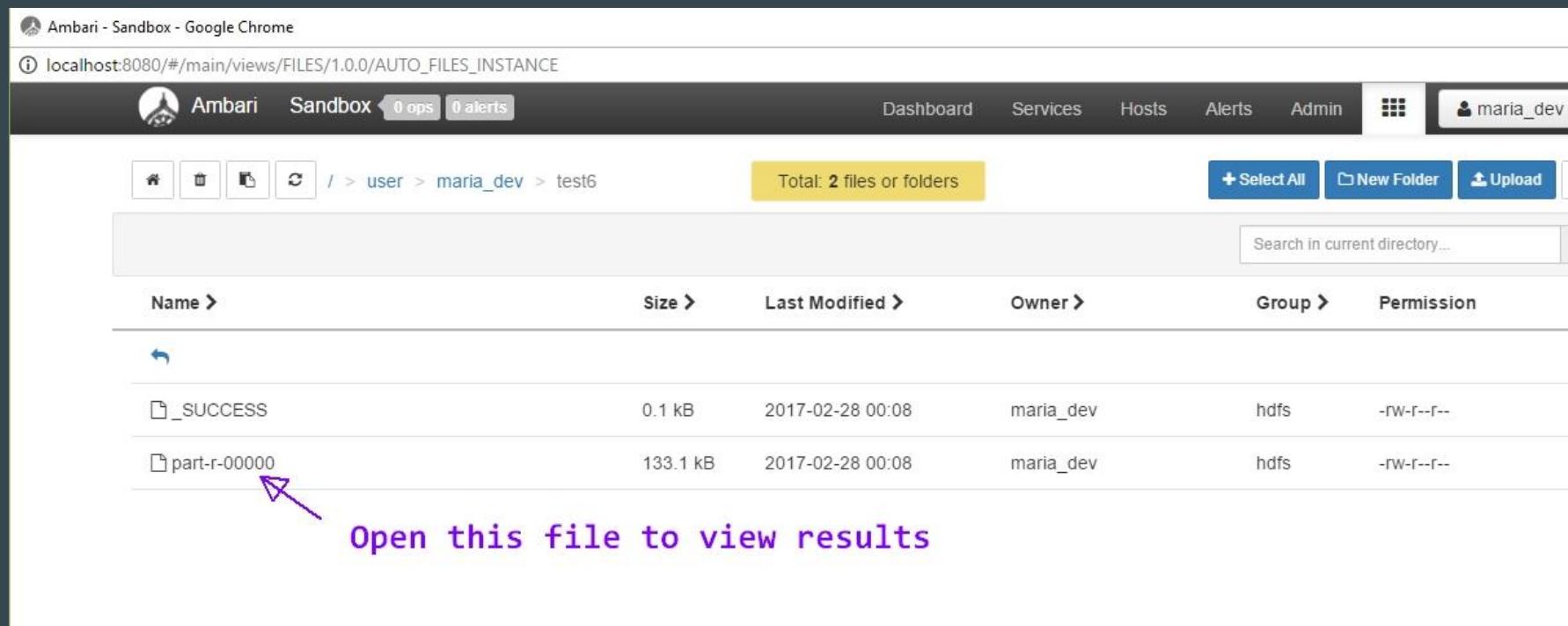
Total: 2 files or folders

+ Select All New Folder Upload

Search in current directory...

Name >	Size >	Last Modified >	Owner >	Group >	Permission
/_SUCCESS	0.1 kB	2017-02-28 00:08	maria_dev	hdfs	-r--r--r--
part-r-00000	133.1 kB	2017-02-28 00:08	maria_dev	hdfs	-r--r--r--

Open this file to view results



File Preview



/user/maria_dev/test6/part-r-00000

```
85048 3.5801528
85050 3.840708
85051 3.2222223
85053 3.516892
85054 3.7346938
85058 4.0
85060 4.125
85064 4.5
85066 4.5
85067 3.6666667
85068 4.6666665
85070 4.0
85072 1.0
85073 3.2727273
85075 2.5
85078 4.75
85080 4.1666665
85082 4.5
85083 4.4333334
85085 3.6394231
```

Cancel

Download

Improve MapReduce Performance

- Increase the number of reducers for the job. But, why?
 - It makes the reducer phase shorter because you get more parallelism
 - Hadoop does this automatically if the map output > HDFS block size
 - Number of reducers can be set manually (shown in next slide)
- Map jobs should run longer than the reducer
 - If map job time is greater than or equal, that is a sign the MapReduce job is not correctly optimized, or input data is very small (as was the case with my example).
 - Remember, map output gets partitioned. The number of partitions is equal to the number of reducers. Very large map output increases the number of partitions.

Improve MapReduce Performance (continued)

- Does very large Map input data mean Hadoop will spawn more reducers?
 - No! The **output** of the map job governs the number of reducers, not the map input data
 - Depending on the query, map functions may produce relatively small output data
 - Larger map input data only means more map tasks will be created by Hadoop.
- How do you set the number of reducers?
 - The MapReduce API will try and derive the correct number of reducers to use
 - However, you can set it manually `job.setNumReduceTasks(25);`
 - Too many reduce tasks can cause problems too. You must fine tune the amount.

Improve MapReduce Performance (continued)

- How many reducers do you choose?
 - One, rule of thumb is to aim for reducers that each run for about 5 minutes, and which produce at least one HDFS block's worth of output (This assumes you're working with 'Big Data')
- Remember, input data to Map is expected to be much larger than my example
 - Increasing the number of reducers for my example would be a bad idea since my input data was only 14 mb, well below the 128 mb block size.

Example of what not to do

My input data was very small but I will increase my reducers to 25 anyway

I added the following code to my job runner in the main function:

```
job.setNumReduceTasks(25);
```

I then transferred the JAR file to hadoop and ran the job again. Let's see if I truly killed MapReduce performance.

Example of what not to do (continued)

```
HDFS: Number of write operations=50
Job Counters
    Launched map tasks=1
    Launched reduce tasks=25 ← Now I have 25 Reducers
    Data-local map tasks=1
    Total time spent by all maps in occupied slots (ms)=5486
    Total time spent by all reduces in occupied slots (ms)=403716 ← Reducers ran 5X longer
    Total time spent by all map tasks (ms)=5486
    Total time spent by all reduce tasks (ms)=403716
    Total vcore-milliseconds taken by all map tasks=5486
    Total vcore-milliseconds taken by all reduce tasks=403716
    Total megabyte-milliseconds taken by all map tasks=1371500
    Total megabyte-milliseconds taken by all reduce tasks=100929000
Map-Reduce Framework
    Map input records=100001
    Map output records=100001
    Map output bytes=964985
    Map output materialized bytes=1165137
    Input split bytes=137
    Combine input records=0
    Combine output records=0
    Reduce input groups=10560
    Reduce shuffle bytes=1165137
    Reduce input records=100001
    Reduce output records=10560
    Spilled Records=200002
```

Example of what not to do (continued)

The screenshot shows a web-based file browser interface for the Ambari Sandbox. The URL is 127.0.0.1:8080/#/main/views/FILES/1.0.0/AUTO_FILES_INSTANCE. The current path is / > user > maria_dev > test07. A yellow box highlights the path 'test07'. The interface includes standard file operations like Open, Rename, Permissions, Delete, Copy, Move, Download, and Upload. A search bar is also present. The main area displays a table of files:

Name	Size	Last Modified	Owner	Group	Permission
part-r-00011	5.2 kB	2017-03-09 12:33	maria_dev	hdfs	-rW-f--f--
part-r-00000	5.2 kB	2017-03-09 12:33	maria_dev	hdfs	-rW-f--f--
part-r-00013	5.4 kB	2017-03-09 12:33	maria_dev	hdfs	-rW-f--f--
part-r-00014	5.5 kB	2017-03-09 12:33	maria_dev	hdfs	-rW-f--f--
part-r-00015	5.4 kB	2017-03-09 12:33	maria_dev	hdfs	-rW-f--f--
part-r-00016	5.1 kB	2017-03-09 12:33	maria_dev	hdfs	-rW-f--f--
part-r-00017	5.1 kB	2017-03-09 12:33	maria_dev	hdfs	-rW-f--f--
part-r-00018	5.3 kB	2017-03-09 12:33	maria_dev	hdfs	-rW-f--f--
part-r-00019	5.7 kB	2017-03-09 12:34	maria_dev	hdfs	-rW-f--f--
part-r-00020	5.0 kB	2017-03-09 12:34	maria_dev	hdfs	-rW-f--f--
part-r-00021	5.2 kB	2017-03-09 12:34	maria_dev	hdfs	-rW-f--f--
part-r-00022	5.5 kB	2017-03-09 12:34	maria_dev	hdfs	-rW-f--f--
part-r-00023	5.2 kB	2017-03-09 12:34	maria_dev	hdfs	-rW-f--f--
part-r-00024	5.0 kB	2017-03-09 12:34	maria_dev	hdfs	-rW-f--f--

A purple arrow points to the last row of the table, highlighting the file 'part-r-00024'.

MapReduce job produced 24 part files, each only ~5kB. That's a lot of small files. This is not good!

Job produced 24 part files, each of which is ~ 5kB, well below the default block size of 128 mb.

That is a lot of unnecessary disk operations

When running a MapReduce job on actual 'Big Data' - make sure the part files are approximately the size of the HDFS block files

Scaling Out Example

Now, I will demonstrate a MapReduce job that works on much larger input data

Query:

Find all businesses that have over 1000 reviews and have a star rating greater than 4

Input data is review3.txt, which is 2.1 GB

Since my input is larger, I expect Hadoop to spawn more map tasks for increased parallelism

Scaling Out Example (continued) MySQL Solution

```
mysql> SELECT business_id, AVG(stars) AS starAVG, COUNT(*) AS reviewCount
-> FROM review
-> GROUP BY business_id
-> HAVING reviewCount > 1000 AND starAVG > 4 LIMIT 25;
+-----+-----+
| business_id | starAVG | reviewCount |
+-----+-----+
| -9e1ONYQuAa-CB_Rrw7Tw | 4.0953 | 1311 |
| -ed0Yc9on37RoIoG2ZgxBA | 4.0045 | 1117 |
| 0FUtilsQrJITLhgDPxLumEw | 4.1688 | 1410 |
| 0NmTwqYEQiKErDv4a55obg | 4.1138 | 1028 |
| 0W4lkclzZThpx3V65bVgig | 4.0674 | 1603 |
| 2iTsrQsPGRH11i1vRvkQ | 4.3959 | 1225 |
| 3BCsAgo_1i4xMuTyLkMLRQ | 4.2906 | 1311 |
| 3GEEy7RP6e4bT4LAiWFMFQ | 4.2161 | 1462 |
| 3kdS15mo9dWC4clrjEDGg | 4.4659 | 2303 |
| 3I54GTr8-E3XPbIxnf_sAA | 4.2856 | 1390 |
| 3N9U549Zse8UP-MwKZAjAQ | 4.2307 | 1036 |
| 4JNXUYY8wbaaDmk3BPz1hw | 4.1140 | 6414 |
| 4k3R1MAMd46DZ_JyZU01Mg | 4.1647 | 1008 |
| 5shgJB7a-2_gdnzc0gs0tg | 4.2727 | 1313 |
| 7sb2FYLS2sejZKxRYF9mtg | 4.4955 | 1443 |
| 7sPNbCx7vGAaH7SbNPZ6oA | 4.0658 | 2735 |
| 9a3DrZvpYxVs3k_qwlCNSw | 4.2656 | 1457 |
| A-uZAD4zP3rRxba44WUGV5w | 4.6825 | 1134 |
| aLcFhMe6DDJ430ze1Cpd2A | 4.0323 | 1145 |
| awI4hHMfa7H0Xf0-ChU5hg | 4.4437 | 1785 |
| BH9z7IJ4zydAqgwbsbqoVZQ | 4.3463 | 1311 |
| CiYLq33nAyghFkUR15pP-Q | 4.3154 | 1151 |
| d10lxZPirV1l0SpdRZJczA | 4.3515 | 1391 |
| DkYS3arLohA8s15uUEmHOw | 4.2668 | 4655 |
| eLFFWcdb7VkjNyTONksHiQ | 4.2761 | 1619 |
+-----+
25 rows in set (36.95 sec)
```

As before, here how you would solve the query using MySQL

Scaling Out Example (continued)

Ambari - Sandbox - Google Chrome
127.0.0.1:8080/#/main/views/FILES/1.0.0/AUTO_FILES_INSTANCE

/ > user > maria_dev > testdata

Total: 7 files or folders

+ Select All

Search

Name >	Size >	Last Modified >	Owner >	Group
business.csv	14.8 MB	2017-02-25 12:45	maria_dev	hdfs
business_attributes.csv	5.1 MB	2017-02-25 13:00	maria_dev	hdfs
review.csv	74.7 MB	2017-02-25 13:14	maria_dev	hdfs
review2.txt	732.7 MB	2017-03-09 15:50	maria_dev	hdfs
review3.txt	2.1 GB	2017-03-09 19:48	maria_dev	hdfs
tip.csv	13.0 MB	2017-02-25 16:26	maria_dev	hdfs
user.csv	11.9 MB	2017-02-25 16:17	maria_dev	hdfs

The input Data is 2.1 GB

Scaling Out Example (continued)

```
Problems @ Javadoc Declaration Console 
terminated> BusinessRating [Java Application] C:\Program Files (x86)\Java\jre1.8.0_111\bin\javaw.exe (Mar 10, 2017, 10:15:34 AM)
<terminated> BusinessRating [Java Application] C:\Program Files (x86)\Java\jre1.8.0_111\bin\javaw.exe (Mar 10, 2017, 10:15:34 AM)
|review_id|||user_id|||business_id|||stars|||date|||text|||useful|||funny|||cool|||type
NxL8SIC5q0dn1XCg18Bg|||Kpk0kG6RIf4Ra25Lhhxf1A|||2aFiy99vNLk1Cx3T_tGS9A|||5|||2011-10-10|||If you enjoy service by someone who is as competent as he is personable, I would recom
pxBbIg0XvLuTi_SPs1hQE0|||b07fQqlotn9hKX-gXRsrsgA|||2aFiy99vNLk1Cx3T_tGS9A|||5|||2010-12-29|||After being on the phone with Verizon Wireless trying to figure out why my phone wasn't
-Rachel Charlupski
Founder and CEO, The Babysitting Company|||1|||0|||0|||review
ws1W2Lu4NYylbjEapAGSw|||r1NUhdNmL6yU9Bn-Yx6FTw|||2aFiy99vNLk1Cx3T_tGS9A|||5|||2011-04-29|||Great service! Corey is very service oriented. Works fast and very efficient with his
GP6YEearUMrzPtQYSF1vVg|||aW3ix1KNzAv0M8q-WghA3Q|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2014-07-14|||Highly recommended. Went in yesterday looking for a dresser to use as a tv stand. Foun
25RlYGq2s5qShi-pn3ufVA|||Y0o-Cip8HqvKp_p9nEGphw|||2LfIuF3_sX6uve-IR-P0jQ|||4|||2014-01-15|||I walked in here looking for a specific piece of furniture. I didn't find it, but wha
Uf1K1iyH_JDKhLvn2e4FQ|||bg13j8yJcRO-00NkUYsXGQ|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2013-04-28|||What a great place! Modern on Melrose has amazing furniture at great prices. I highly
oFMVzh-La7SuvpHrH_A14Q|||CWFK9de-nskLYEqDDCfubg|||2LfIuF3_sX6uve-IR-P0jQ|||4|||2014-10-12|||A hidden gem! Found a beautiful buffet for a great price. Whether you're looking for n
brdVt88Mj_YMT1LbjDLxQ|||GJ7PTY7huYORFKKg3db3Gw|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2012-09-18|||This place is a great for those vintage/mid century modern finds. From clothing to cou
zNUSxqf1ZKgKD1NQH3jdFA|||rxqp9eXZj1jYTn0Uism3Hg|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2015-10-11|||This is the place to go for all your Mid Century finds.
We live in Oro Valley and it is not too far to travel when we need a specific item.
I called yesterday asking if they have something we need and minutes later I was sent numerous pictures. Because of the excellent staff, I will be going up there today.
One of my number one places to go when I go to Phoenix.
Prices are very fair for these past treasures.|||0|||0|||0|||review
LkP1l7sZiW0V6IKNLqOp_A|||UU0nHQtHPMAFLidk8tOHTg|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2015-04-05|||Great items at a good price. Helpful, easy to deal with owners. Just the sort of pla
MvvT0BtQHwq7K-pPgkoEQ|||A_Hyfk3FcwFViK1QC7z7w|||2LfIuF3_sX6uve-IR-P0jQ|||1|||2014-07-08|||Disappointing. I've been there twice hoping that a re-return would change my original
It's just an unwelcoming place. So much potential, but the attitude there is less than acceptable. Will spend my $$$ elsewhere.|||1|||0|||0|||review
djGuzIfNkTgKk8ry0PdFQ|||0vD92wp0-uuFoGLBymwfKQ|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2014-08-23|||I'm a local Real Estate agent and have fallen in love with all of the AMAZZZING Mid-
After going in a bunch of places I can say that I really love this store and can't wait to use them to stage my newest listing in a few months at 24th and Indian School.
Let me know if you need help finding a cool house with character to put your amazing finds from Modern on Melrose.
Enjoy Data is not well formed. To ensure good output data quality the map function will need to do extra work compared to the last MapReduce job.
Steven
480-370-5570
stevenonmill@gmail.com|||1|||1|||1|||review
L_yb0QTE7WTg5k0cD8vsA|||END_cFmY8PDTwz82H_MVw|||2LfIuF3_sX6uve-IR-P0jQ|||4|||2015-01-12|||Modern on Melrose is one of our favorite shops on La Serna. It is on the noisy side, but
```

Scaling Out Example (continued)

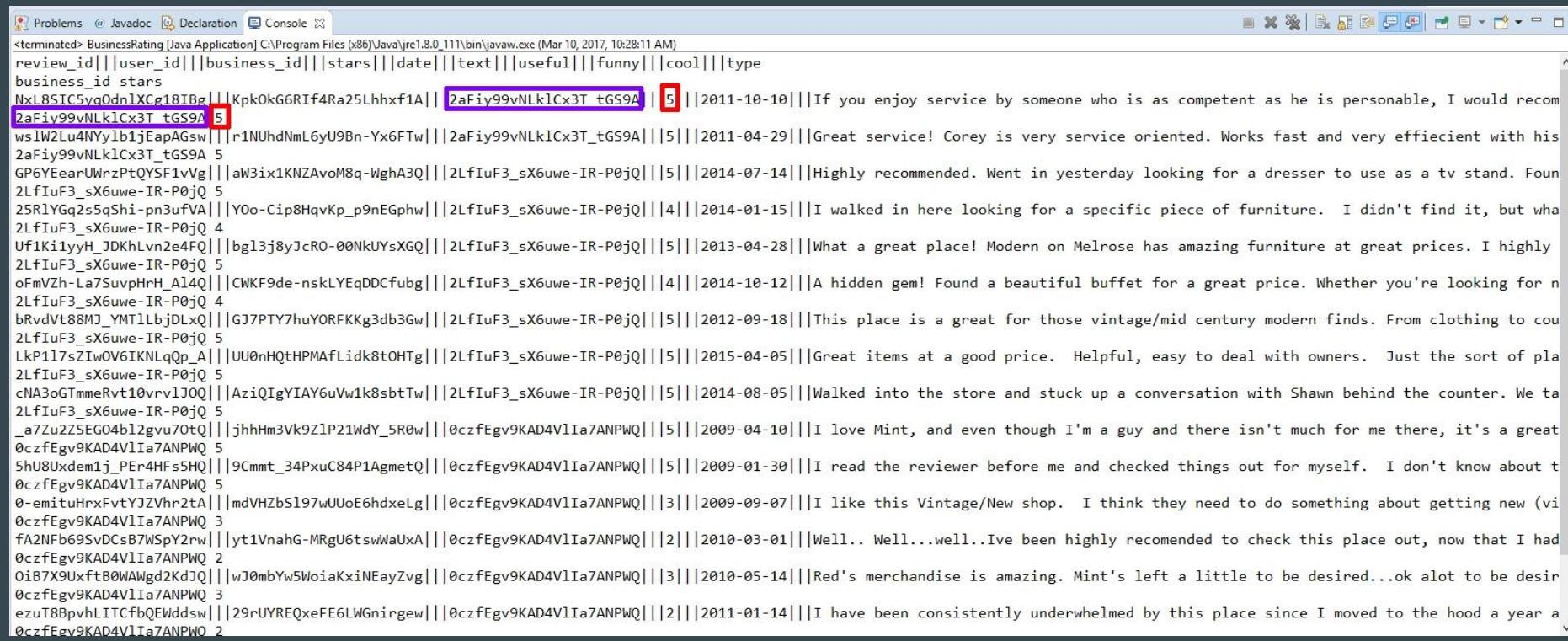
The values in review3.txt were delimited by three pipe symbols |||

First, I identified the target data I would need - business_id, stars

I wrote a java program to help zero in on the target data

I choose only those lines that were well formed and throughout the rest

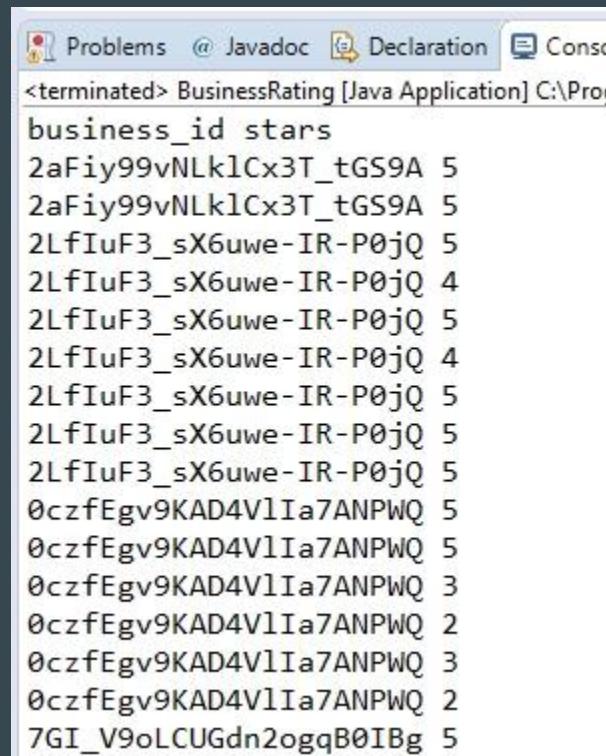
Scaling Out Example (continued) Zero in on target data



The screenshot shows an IDE interface with a Java application named "BusinessRating" running. The code is a simple class with a main method that prints a list of reviews from a file. The reviews are represented as strings containing various fields like review_id, user_id, business_id, stars, date, text, useful, funny, cool, type, and a comment. A specific review is highlighted with a red box around the stars value "5".

```
Problems @ Javadoc Declaration Console
<terminated> BusinessRating [Java Application] C:\Program Files (x86)\Java\jre1.8.0_111\bin\javaw.exe (Mar 10, 2017, 10:28:11 AM)
review_id|||user_id|||business_id|||stars|||date|||text|||useful|||funny|||cool|||type
business_id stars
NxL8SIC5ya0dn1Xe18IBg|||KpkOkG6RIf4Ra25Lhhxf1A|||2aFiy99vNLk1Cx3T_tGS9A|||5|||2011-10-10|||If you enjoy service by someone who is as competent as he is personable, I would recom
2aFiy99vNLk1Cx3T_tGS9A 5
wslW2Lu4NYylb1jEapAGsw|||r1NUhdNmL6yU9Bn-Yx6FTw|||2aFiy99vNLk1Cx3T_tGS9A|||5|||2011-04-29|||Great service! Corey is very service oriented. Works fast and very effiecient with his
2aFiy99vNLk1Cx3T_tGS9A 5
GP6YEarlWlrzPTQYSf1Vg|||aw3ix1KNZAvoM8q-WghA3Q|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2014-07-14|||Highly recommended. Went in yesterday looking for a dresser to use as a tv stand. Foun
2LfIuF3_sX6uve-IR-P0jQ 5
25R1YQg2s5qShi-pn3ufVA|||Yo0-Cip8HqvKp_p9nEGphw|||2LfIuF3_sX6uve-IR-P0jQ|||4|||2014-01-15|||I walked in here looking for a specific piece of furniture. I didn't find it, but wha
2LfIuF3_sX6uve-IR-P0jQ 4
Uf1Ki1yyH_JDKhLvn2e4FQ|||bg13j8yJcR0-00NkUYsXGQ|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2013-04-28|||What a great place! Modern on Melrose has amazing furniture at great prices. I highly
2LfIuF3_sX6uve-IR-P0jQ 5
oFmVZh-La7SuvpHrH_A14Q|||CWKF9de-nskLYEqDDCfubg|||2LfIuF3_sX6uve-IR-P0jQ|||4|||2014-10-12|||A hidden gem! Found a beautiful buffet for a great price. Whether you're looking for n
2LfIuF3_sX6uve-IR-P0jQ 4
bRvdVt88Mj_YMT1LbjDLxQ|||GJ7PTY7huYORFKKg3db3Gw|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2012-09-18|||This place is a great for those vintage/mid century modern finds. From clothing to cou
2LfIuF3_sX6uve-IR-P0jQ 5
LkP117sZ1w0V61KNLqOp_A|||UU0nHQtHPMAfLidk8t0HTg|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2015-04-05|||Great items at a good price. Helpful, easy to deal with owners. Just the sort of pla
2LfIuF3_sX6uve-IR-P0jQ 5
cNA3oGtmmmeRvt10rv1JQ|||AziQIgYIAY6uVw1k8sbtTw|||2LfIuF3_sX6uve-IR-P0jQ|||5|||2014-08-05|||Walked into the store and stuck up a conversation with Shawn behind the counter. We ta
2LfIuF3_sX6uve-IR-P0jQ 5
_a7Zu2ZSEG04b12gvu70tQ|||jhhHm3Vk9Z1P21WdY_5R0w|||0czfEgv9KAD4V1Ia7ANPWQ|||5|||2009-04-10|||I love Mint, and even though I'm a guy and there isn't much for me there, it's a great
0czfEgv9KAD4V1Ia7ANPWQ 5
5hU8Uxdem1j_PEr4HFs5HQ|||9Cmmnt_34PxuC84P1AgmetQ|||0czfEgv9KAD4V1Ia7ANPWQ|||5|||2009-01-30|||I read the reviewer before me and checked things out for myself. I don't know about t
0czfEgv9KAD4V1Ia7ANPWQ 5
0-emituHrxFvtYZVhr2ta|||mdVHZbS197wUuoE6hdxeLg|||0czfEgv9KAD4V1Ia7ANPWQ|||3|||2009-09-07|||I like this Vintage/New shop. I think they need to do something about getting new (vi
0czfEgv9KAD4V1Ia7ANPWQ 3
fa2NFb695vDcsB7WSpY2rw|||yt1VnahG-MRgU6tswWaUxA|||0czfEgv9KAD4V1Ia7ANPWQ|||2|||2010-03-01|||Well.. Well...well..Ive been highly recomended to check this place out, now that I had
0czfEgv9KAD4V1Ia7ANPWQ 2
Oib7X9UxftB0WAwdg2KdQ|||wJ0mbYw5WoiaKxiNEayZvg|||0czfEgv9KAD4V1Ia7ANPWQ|||3|||2010-05-14|||Red's merchandise is amazing. Mint's left a little to be desired...ok alot to be desir
0czfEgv9KAD4V1Ia7ANPWQ 3
ezuT8BpvhLITcfbQEwidsw|||29rUYREQxeFE6LWGnirgew|||0czfEgv9KAD4V1Ia7ANPWQ|||2|||2011-01-14|||I have been consistently underwhelmed by this place since I moved to the hood a year a
0czfEgv9KAD4V1Ia7ANPWQ 2
```

Scaling Out Example (continued)



A screenshot of an IDE interface showing a terminal window. The window title is '<terminated> BusinessRating [Java Application] C:\Prog'. The console tab is active, displaying the following text:

```
business_id stars
2aFiy99vNLk1Cx3T_tGS9A 5
2aFiy99vNLk1Cx3T_tGS9A 5
2LfIuF3_sX6uve-IR-P0jQ 5
2LfIuF3_sX6uve-IR-P0jQ 4
2LfIuF3_sX6uve-IR-P0jQ 5
2LfIuF3_sX6uve-IR-P0jQ 4
2LfIuF3_sX6uve-IR-P0jQ 4
2LfIuF3_sX6uve-IR-P0jQ 5
2LfIuF3_sX6uve-IR-P0jQ 5
2LfIuF3_sX6uve-IR-P0jQ 5
2LfIuF3_sX6uve-IR-P0jQ 5
0czfEgv9KAD4V1Ia7ANPWQ 5
0czfEgv9KAD4V1Ia7ANPWQ 5
0czfEgv9KAD4V1Ia7ANPWQ 3
0czfEgv9KAD4V1Ia7ANPWQ 2
0czfEgv9KAD4V1Ia7ANPWQ 3
0czfEgv9KAD4V1Ia7ANPWQ 2
7GI_V9oLCUGdn2oggB0IBg 5
```

I captured the target data.

This is what my Map function will need to do

Now it is time to write code using the MapReduce API

Scaling Out Example (continued) The Map Function

```
Text business_id = new Text();
FloatWritable stars = new FloatWritable();

public void map(Object key, Text value, Context context)
    throws IOException, InterruptedException {

    String line = value.toString(); // Convert to Java String

    line = line.replace("\\"", ""); // Remove double quotes
    String delimiters = "(\\|\\|\\|\\|)"; // Column values separated by ||
    String[] colValue = line.split(delimiters); // Place values in an array

    // The array should have a size of 10 if it is correct
    int arraySize = colValue.length;

    // extract the third and fourth elements which are
    // business_id and stars respectively
    if(arraySize == 10){
        business_id.set(colValue[2]); // assign the 3rd element to business_id
        try {
            stars.set(Float.valueOf(colValue[3])); // assign 4th element to stars
        }
        catch (NumberFormatException e) {
            e.printStackTrace();
        }
    }

    // Write output key, value pair to the context object
    context.write(business_id, stars);
}
```

Scaling Out Example (continued) The Reduce Function

```
FloatWritable business_rating = new FloatWritable();

public void reduce(Text key, Iterable<FloatWritable> values, Context context)
    throws IOException, InterruptedException {
    // Record sum and count of stars
    float sum = 0;
    float count = 0;
    // Sum all the stars that share the same postal code
    for (FloatWritable val : values) {
        sum += val.get();
        count++;
    }
    // Compute the average
    float avgRating = (sum/count);

    // Only want business with over 1000 reviews and a rating > 4
    if(count > 1000 && avgRating > 4){
        business_rating.set(avgRating);
        // Write output pair
        context.write(key, business_rating);
    }
}
```

Scaling Out Example (continued) The Job Runner

```
// The Driver Code
public static void main(String[] args) throws Exception {
    // Check for correct number of arguments
    if (args.length != 2) {
        System.out.println("usage: [input] [output]");
        System.exit(-1);
    }
    // Create and configure MapReduce job
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "BusinessRating");
    job.setJarByClass(BusinessRating.class);
    // Set Mapper and Reducer Class
    //job.setNumReduceTasks(25); // Explicitly set number of reducers
    job.setMapperClass(MapperClass.class);
    job.setReducerClass(ReducerClass.class);
    // Set output types for keys and values
    job.setMapOutputKeyClass(Text.class);
    job.setMapOutputValueClass(FloatWritable.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(FloatWritable.class);
    // Set input and output format class
    job.setInputFormatClass(TextInputFormat.class); //default
    job.setOutputFormatClass(TextOutputFormat.class);
    // Set input and output files from arguments
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    // Send job to cluster and wait for job to complete
    job.waitForCompletion(true);
}
```

Scaling Out Example (continued) The Results

File System Counters

```
FILE: Number of bytes read=310473543  
FILE: Number of bytes written=623702148  
FILE: Number of read operations=0  
FILE: Number of large read operations=0  
FILE: Number of write operations=0  
HDFS: Number of bytes read=2301147471  
HDFS: Number of bytes written=20238  
HDFS: Number of read operations=57  
HDFS: Number of large read operations=0  
HDFS: Number of write operations=2
```

Job Counters

```
Launched map tasks=18  
Launched reduce tasks=1  
Data-local map tasks=18  
Total time spent by all maps in occupied slots (ms)=837885  
Total time spent by all reduces in occupied slots (ms)=78331  
Total time spent by all map tasks (ms)=837885  
Total time spent by all reduce tasks (ms)=78331  
Total vcore-milliseconds taken by all map tasks=837885  
Total vcore-milliseconds taken by all reduce tasks=78331  
Total megabyte-milliseconds taken by all map tasks=209471250  
Total megabyte-milliseconds taken by all reduce tasks=19582750
```

Map-Reduce Framework

```
Map input records=10706094  
Map output records=10706094  
Map output bytes=289061337  
Map output materialized bytes=310473633  
Input split bytes=2448  
Combine input records=0
```

Since input data was much larger this time, it was divided into block size chunks - One Map Task per block

Better time ratio

Scaling Out Example (continued) The Results

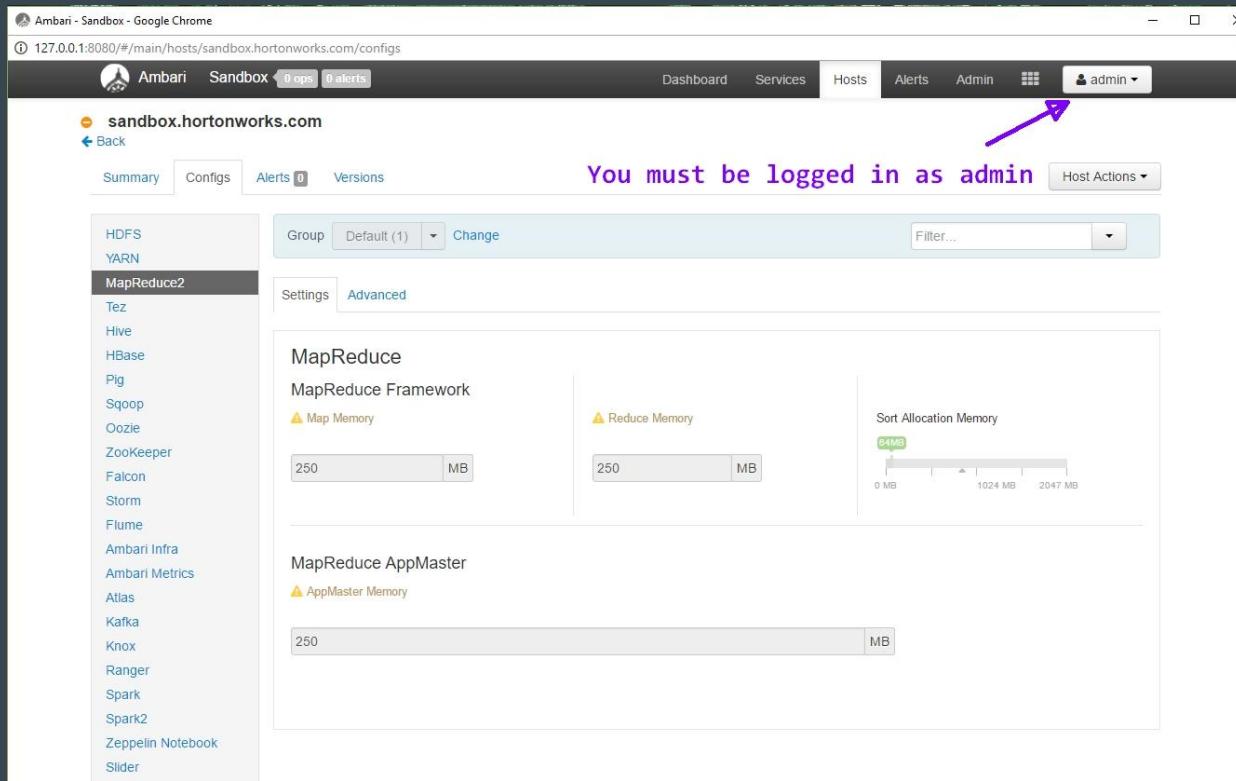
File Preview
/user/maria_dev/test14/part-r-00000

```
--9e10NYQuAa-CB_Rrw7Tw 4.143018
-050d_XIor1NpCuLkbIVaQ 4.0801916
-1xuC540Nyght_iWFeJ-dw 4.3288083
-3zffZUHoY8bQjGfPSoBKQ 4.196052
-4TMQnQJW1yd6NqGRDvAeA 4.0705705
-95mbLJsa0CxXhpaNL4LvA 4.1923323
-9dmhyBvepc08KPEH1EM0w 4.135856
-Dnh48f029YNugtMKkkI-Q 4.404682
-Eu04UHRqmGGyvRDY8-tg 4.6044354
-Ht7HiGBoxx8lS1Y8IPj08g 4.0825796
-ICGmF2qUVKdvOehVNgPbg 4.10951
-Le6cwbZL4tDZwNHwipfKg 4.55541
-0xDX2fPQLYi6ChW2Z6xxQ 4.191556
-WLrZPzjkKfrftLWaCi1QZQ 4.001311
-bMZCfTK7fxFaURynKpBMA 4.288835
-yApKLEFAvvNyifvpNKWCA 4.2094216
-yQHIYKXH3HAdhh1W520MQ 4.3481483
01fuY2NNscttoTx0YbuZXw 4.17973
07AZL5XenC0 -op_onKLdw 4.0614643
```

Job output produced one part file.
It's size is 19.8 KB. There was no need to produce another part file since this is well below HDFS block size.

Output file shows all the businesses that have over 1000 reviews and have a rating greater than 4

Boosting Performance Using Ambari GUI



Ambari - Sandbox - Google Chrome
127.0.0.1:8080/#/mainhosts/sandbox.hortonworks.com/configs

Ambari Sandbox 0 ops 0 alerts

sandbox.hortonworks.com

You must be logged in as admin

Summary Configs Alerts 0 Versions

Group Default (1) Change Filter...

MapReduce2

Settings Advanced

MapReduce

MapReduce Framework

- Map Memory: 250 MB
- Reduce Memory: 250 MB
- Sort Allocation Memory: 84MB (Slider from 0 MB to 2047 MB)

MapReduce AppMaster

- AppMaster Memory: 250 MB

A sidebar on the left lists services: HDFS, YARN, MapReduce2, Tez, Hive, HBase, Pig, Sqoop, Oozie, ZooKeeper, Falcon, Storm, Flume, Ambari Infra, Ambari Metrics, Atlas, Kafka, Knox, Ranger, Spark, Spark2, Zeppelin Notebook, Slider.

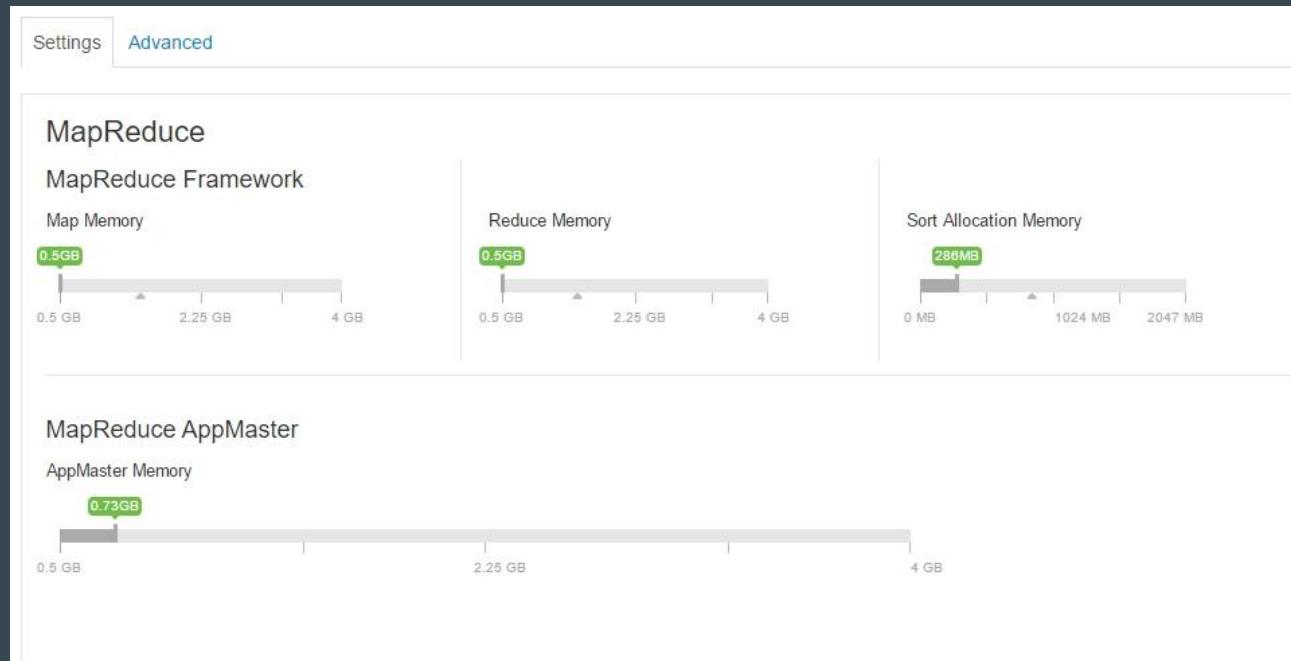
You must login to Ambari as the administrator

Notice that all the MapReduce memory settings are at the default of 250 MB.

This value is below the recommended value.

I will boost performance by increasing this value into the recommended range.

Boosting Performance Using Ambari GUI (continued)



Now the settings are in the recommended range.

I had to restart the MapReduce service and then I ran my MapReduce job again.

Boosting Performance Using Ambari GUI (continued)

```
HDFS: Number of write operations=2
Job Counters
  Launched map tasks=18
  Launched reduce tasks=1
  Data-local map tasks=18
  Total time spent by all maps in occupied slots (ms)=774192
  Total time spent by all reduces in occupied slots (ms)=82228
  Total time spent by all map tasks (ms)=774192
  Total time spent by all reduce tasks (ms)=82228
  Total vcore-milliseconds taken by all map tasks=774192
  Total vcore-milliseconds taken by all reduce tasks=82228
  Total megabyte-milliseconds taken by all map tasks=193548000
  Total megabyte-milliseconds taken by all reduce tasks=20557000
Map-Reduce Framework
  Map input records=10706094
  Map output records=10706094
  Map output bytes=289061337
  Map output materialized bytes=310473633
  Input split bytes=2448
  Combine input records=0
  Combine output records=0
  Reduce input groups=110740
  Reduce shuffle bytes=310473633
  Reduce input records=10706094
  Reduce output records=624
  Spilled Records=21412188
  Shuffled Maps =18
  Failed Shuffles=0
  Merged Map outputs=18
  GC time elapsed (ms)=45055
  CPU time spent (ms)=213130
  Physical memory (bytes) snapshot=3486101504
  Virtual memory (bytes) snapshot=36732022784
  Total committed heap usage (bytes)=2268594176
```

CPU time spent before optimization 213130 ms



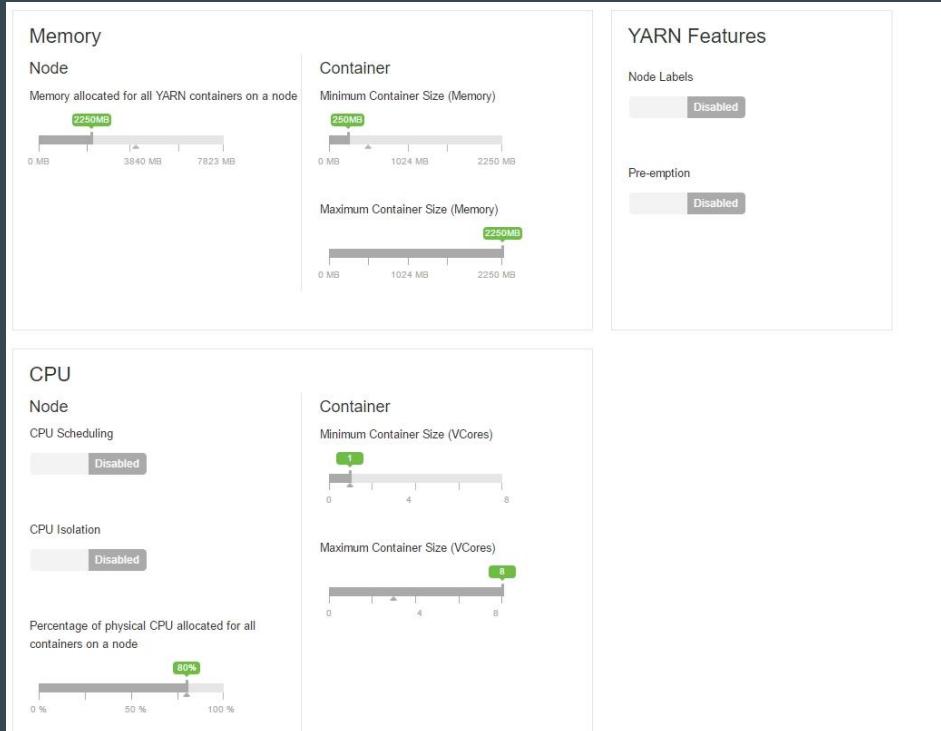
Boosting Performance Using Ambari GUI (continued)

```
Total megabyte-milliseconds taken by all map tasks=115046912  
Total megabyte-milliseconds taken by all reduce tasks=32114688  
  
Map-Reduce Framework  
Map input records=10706094  
Map output records=10706094  
Map output bytes=289061337  
Map output materialized bytes=310473633  
Input split bytes=2448  
Combine input records=0  
Combine output records=0  
Reduce input groups=110740  
Reduce shuffle bytes=310473633  
Reduce input records=10706094  
Reduce output records=624  
Spilled Records=21412188  
Shuffled Maps =18  
Failed Shuffles=0  
Merged Map outputs=18  
GC time elapsed (ms)=9178  
CPU time spent (ms)=168230  
Physical memory (bytes) snapshot=9801912320  
Virtual memory (bytes) snapshot=53080670208  
Total committed heap usage (bytes)=9102688256  
  
Shuffle Errors  
BAD_ID=0  
CONNECTION=0  
IO_ERROR=0  
WRONG_LENGTH=0  
WRONG_MAP=0  
WRONG_REDUCE=0  
  
File Input Format Counters  
Bytes Read=2301145023  
File Output Format Counters  
Bytes Written=20238
```

Previous 213130 ms
Current 168230 ms
22% Faster!

Wow, a small change caused a big boost in performance!

Scaling UP

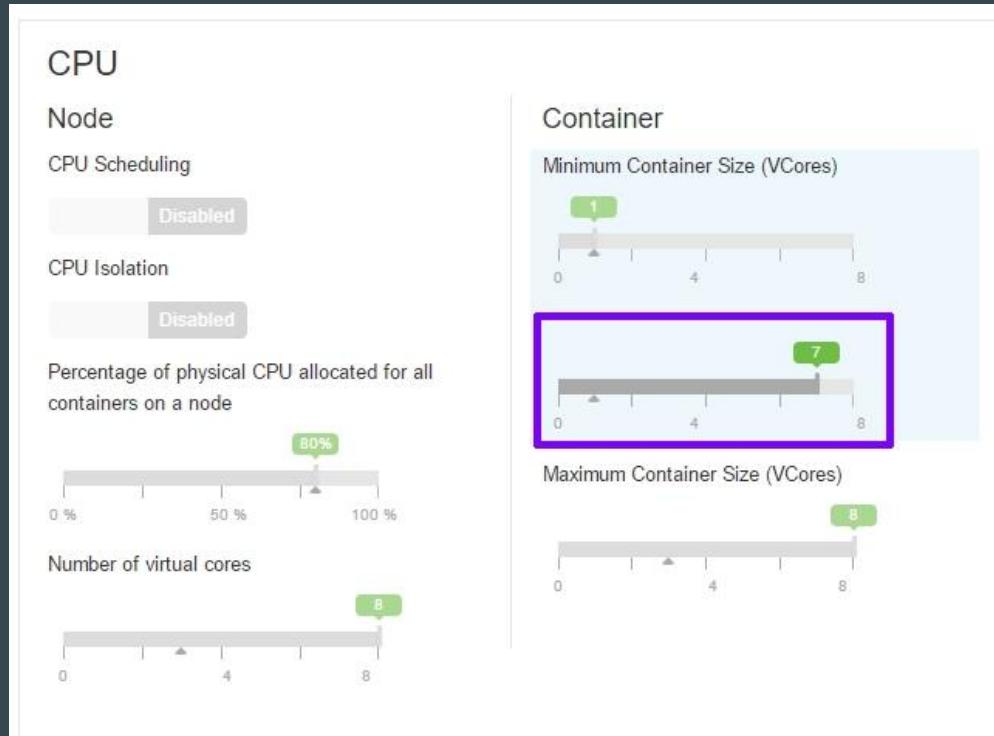


Minimum virtual cores = 1

I will increase to 7 and run the job again.

Let's see if this improves performance of the MapReduce job

Scaling Up



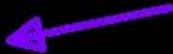
I increased the VCores amount.

Now Minimum VCores = 7

Scaling Up - After increasing min. VCore amount

```
total vcore-milliseconds taken by all reduce tasks=01572
Total megabyte-milliseconds taken by all map tasks=115852288
Total megabyte-milliseconds taken by all reduce tasks=31524864
Map-Reduce Framework
  Map input records=10706094
  Map output records=10706094
  Map output bytes=289061337
  Map output materialized bytes=310473633
  Input split bytes=2448
  Combine input records=0
  Combine output records=0
  Reduce input groups=110740
  Reduce shuffle bytes=310473633
  Reduce input records=10706094
  Reduce output records=624
  Spilled Records=21412188
  Shuffled Maps =18
  Failed Shuffles=0
  Merged Map outputs=18
  GC time elapsed (ms)=8283
  CPU time spent (ms)=165680
  Physical memory (bytes) snapshot=9689694208
  Virtual memory (bytes) snapshot=53111750656
  Total committed heap usage (bytes)=9175564288
Shuffle Errors
  BAD_ID=0
  CONNECTION=0
  IO_ERROR=0
  WRONG_LENGTH=0
  WRONG_MAP=0
  WRONG_REDUCE=0
File Input Format Counters
```

Before Optimization 168230 ms
After Optimization 165680 ms
Negligible improvement due to relatively small MapReduce job



Improvements

Write a **Combiner** function as an optimization

The combiner function would work on the output of the Map function

It could sum the star ratings and feed reducer the sum and number of businesses

Optimized reducer input < 94582, [345, 75] >

Sum of stars number of businesses

The reducer would only have to compute the average for each postal code

Maintain Data Quality

- My MapReduce job worked on clean, historical data
- At Enterprise level, new ‘dirty’ data coming in all the time
- Remember, garbage in = garbage out
- Veracity - the 4th V of Big Data
- Turn Big Data into smaller data by filtering and cleaning
- Preprocessing data before inputting into MapReduce job ensures high quality data output

How do you maintain data integrity?

- ❑ Preprocessing (cleaning, filtering) does not necessarily ensure data integrity
- ❑ Can't impose tuple based constraints or attribute constraints like with MySQL
- ❑ Solution -
 - ❑ Determine the data type violations that may cause data integrity problems
 - ❑ Further parse data in Map function to check for data type violations

Example: In the map function it may be a good idea to check that the star ratings are numeric and non-negative.

Other examples: In map function, by using regular expressions check that a phone number, zip code, or age attributes are valid.

Optimization Using File Compression

Two benefits of using file compression:

1. Reduces the space required to store data files
2. Speeds up data transfer across network or reading to/from disk

If Input Data Files are Compressed

- The bytes read from HDFS is reduced
- They will be decompressed automatically by MapReduce job

Infrequently Viewed Job Output Data Should Be Stored In Compressed Format

- Saves disk space

Compress Map Output Data (Intermediate Compression)

- Jobs run faster by reducing the volume of data transferred to the Reducer Functions

Warning!

Use a compression format the supports file splitting!

If you do not the MapReduce job will be very inefficient.

Data locality optimization will be lost!

To Enable gzip Map Output Compression (intermediate compression)

In the main function and the following to the configuration settings:

```
Configuration conf = new Configuration();

conf.setBoolean(Job.MAP_OUTPUT_CORESS, true);

conf.setClass(Job.MAP_OUTPUT_COMPRESS_CODEC, GzipCodec.class, \
CompressionCodec.class);

Job job = new Job(conf);
```

What I learned

Writing MapReduce jobs is not easy!

Task decomposition can be tricky

Understand the inputs and outputs before coding

Functional program thinking

You must remember what Hadoop does for you (grouping and sort)

I'd prefer to work at a higher-level of abstraction

References

- Hortonworks Sandbox
 - [Getting Started with HDFS Sandbox](#)
 - [Learning the Ropes of the Hortonworks Sandbox](#)
 - [Manage Files on HDFS via CLI/Ambari Files View](#)
- MapReduce
 - [Introducing Apache Hadoop to Java Developers](#)
 - [Apache's MapReduce Tutorial](#)
 - [Apache Hadoop API Documentation](#)
 - Hadoop The Definitive Guide (4th Edition) by Tom White ← Very Good Book