Single Node Cluster Setup with Cloudera Distributed Hadoop (CDH)

Big Data (CS522)

Maharishi University of Management

Submitted By

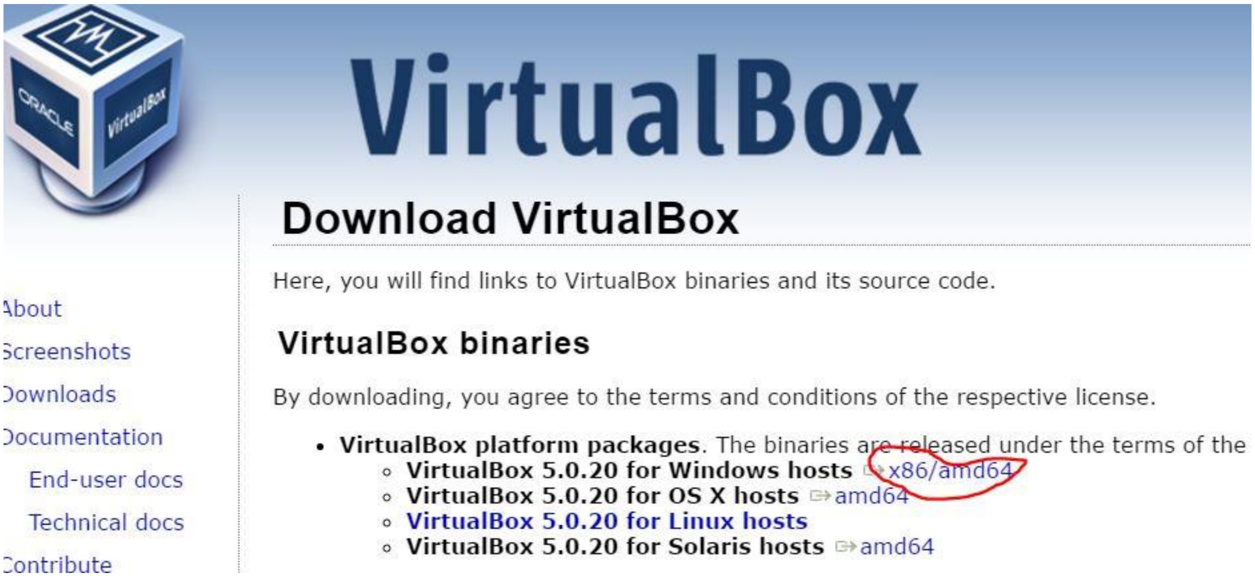
Jivan Nepali, 985095

Jun 06, 2016

Download and Install Oracle VirtualBox

Use the following link to download the VirtualBox setup file:

https://www.virtualbox.org/wiki/Downloads



VirtualBox

Download VirtualBox

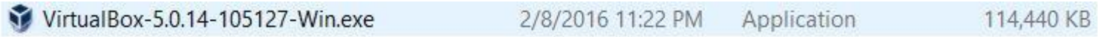
Here, you will find links to VirtualBox binaries and its source code.

VirtualBox binaries

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

- **VirtualBox platform packages**. The binaries are released under the terms of the
  - **VirtualBox 5.0.20 for Windows hosts** x86/amd64
  - **VirtualBox 5.0.20 for OS X hosts** amd64
  - **VirtualBox 5.0.20 for Linux hosts**
  - **VirtualBox 5.0.20 for Solaris hosts** amd64

The version, I downloaded, was 5.0.14 –

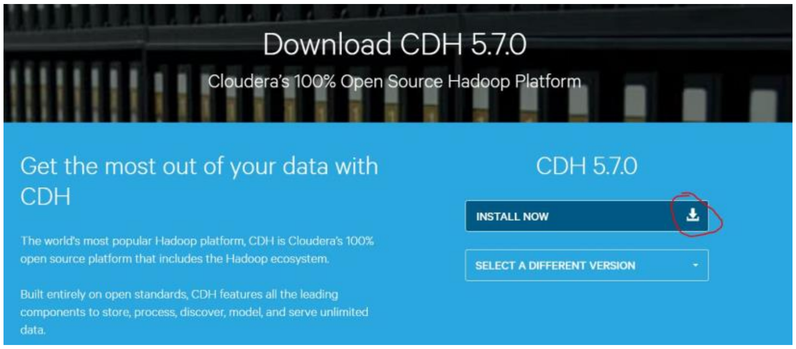


VirtualBox-5.0.14-105127-Win.exe	2/8/2016 11:22 PM	Application	114,440 KB

Download and Install Cloudera Distributed Hadoop (CDH)

Use the following link:

http://www.cloudera.com/downloads/cdh/5-7-0.html



Download CDH 5.7.0

Cloudera's 100% Open Source Hadoop Platform

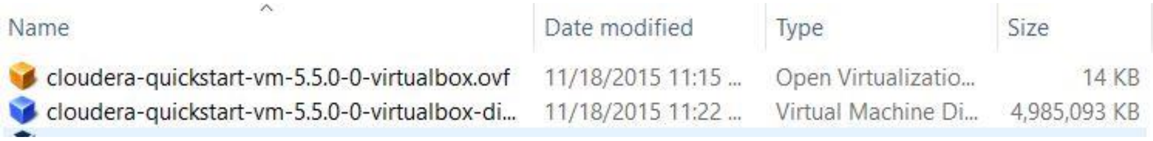
Get the most out of your data with CDH

The world's most popular Hadoop platform, CDH is Cloudera's 100% open source platform that includes the Hadoop ecosystem.

Built entirely on open standards, CDH features all the leading components to store, process, discover, model, and serve unlimited data.

CDH 5.7.0

The VM file is about 5 GB, so have patience before it gets completely downloaded! We can download the latest stable version. I'd already downloaded the CDH5.5 version –

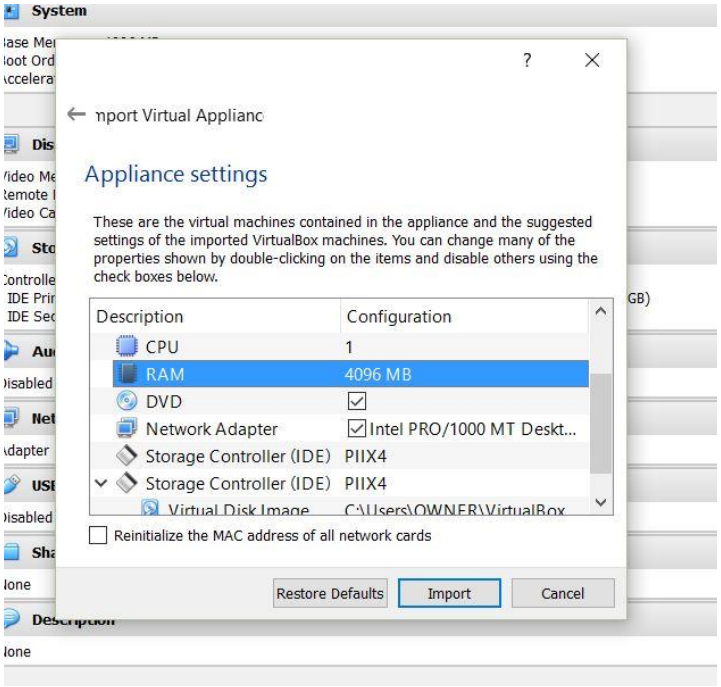


Name	Date modified	Type	Size
cloudera-quickstart-vm-5.5.0-0-virtualbox.ovf	11/18/2015 11:15 ...	Open Virtualizatio...	14 KB
cloudera-quickstart-vm-5.5.0-0-virtualbox-di...	11/18/2015 11:22 ...	Virtual Machine Di...	4,985,093 KB

Configure VM

Import CDH VM into VirtualBox

In the VirtualBox, go to File > Import Appliance, select correct appliance, then click Next and Import

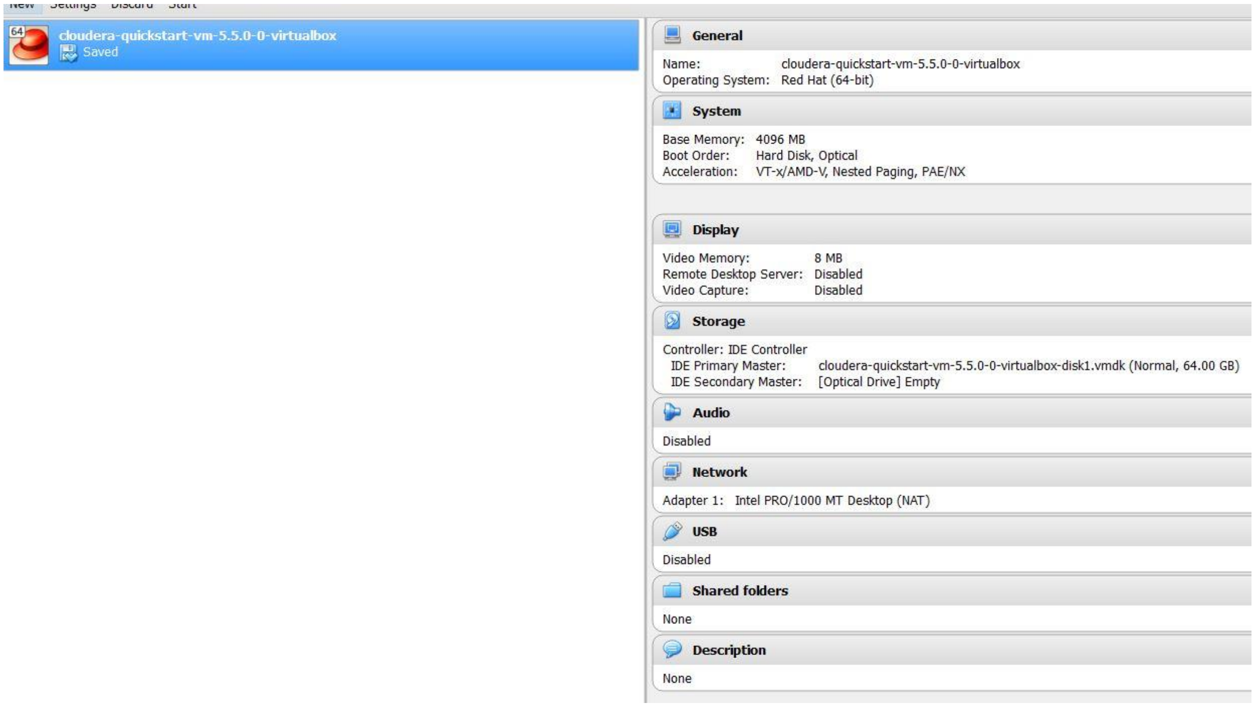


Appliance settings

These are the virtual machines contained in the appliance and the suggested settings of the imported VirtualBox machines. You can change many of the properties shown by double-clicking on the items and disable others using the check boxes below.

Description	Configuration
CPU	1
RAM	4096 MB
DVD	☑
Network Adapter	☑ Intel PRO/1000 MT Deskt...
Storage Controller (IDE)	PIIX4
Storage Controller (IDE)	PIIX4
Virtual Disk Image	C:\Users\OWNER\VirtualBox

□ Reinitialize the MAC address of all network cards



cloudera-quickstart-vm-5.5.0-0-virtualbox
Saved

General

Name: cloudera-quickstart-vm-5.5.0-0-virtualbox
Operating System: Red Hat (64-bit)

System

Base Memory: 4096 MB
Boot Order: Hard Disk, Optical
Acceleration: VT-x/AMD-V, Nested Paging, PAE/NX

Display

Video Memory: 8 MB
Remote Desktop Server: Disabled
Video Capture: Disabled

Storage

Controller: IDE Controller
IDE Primary Master: cloudera-quickstart-vm-5.5.0-0-virtualbox-disk1.vmdk (Normal, 64.00 GB)
IDE Secondary Master: [Optical Drive] Empty

Audio

Disabled

Network

Adapter 1: Intel PRO/1000 MT Desktop (NAT)

USB

Disabled

Shared folders

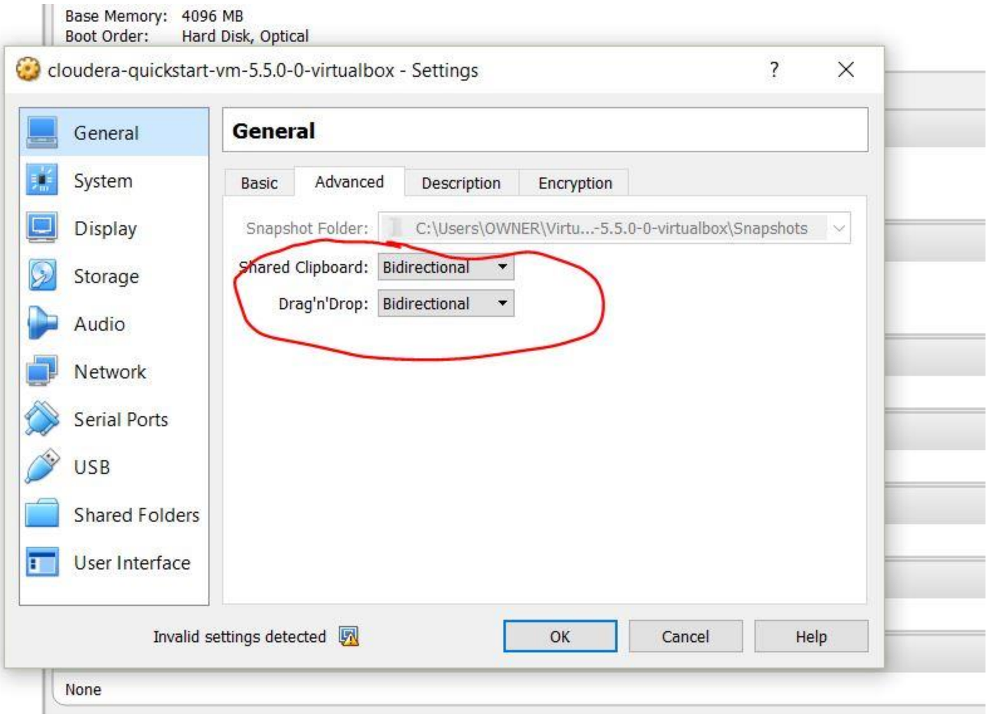
None

Description

None

Configure Settings

Click on Settings menu and configure as like below:



Base Memory: 4096 MB
Boot Order: Hard Disk, Optical

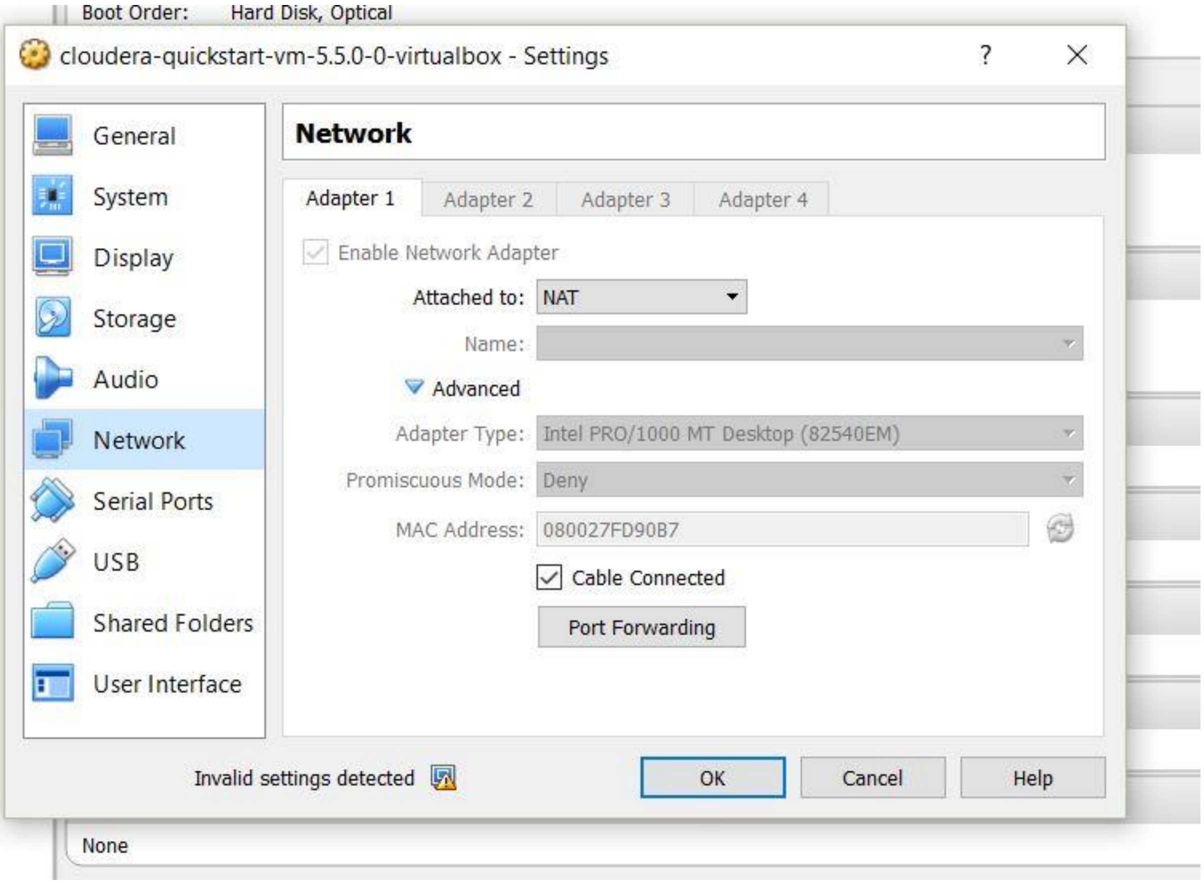
cloudera-quickstart-vm-5.5.0-0-virtualbox - Settings

General

Snapshot Folder: C:\Users\OWNER\Virtu...-5.5.0-0-virtualbox\Snapshots
Shared Clipboard: Bidirectional
Drag'n'Drop: Bidirectional

Invalid settings detected

None



Boot Order: Hard Disk, Optical

cloudera-quickstart-vm-5.5.0-0-virtualbox - Settings

Network

☑ Enable Network Adapter

Attached to: NAT

Name:

Advanced

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Deny

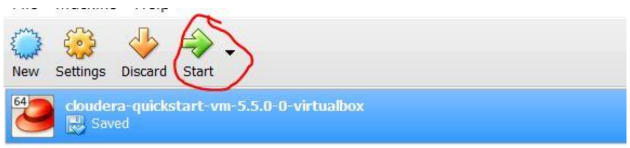
MAC Address: 080027FD90B7

☑ Cable Connected

Invalid settings detected

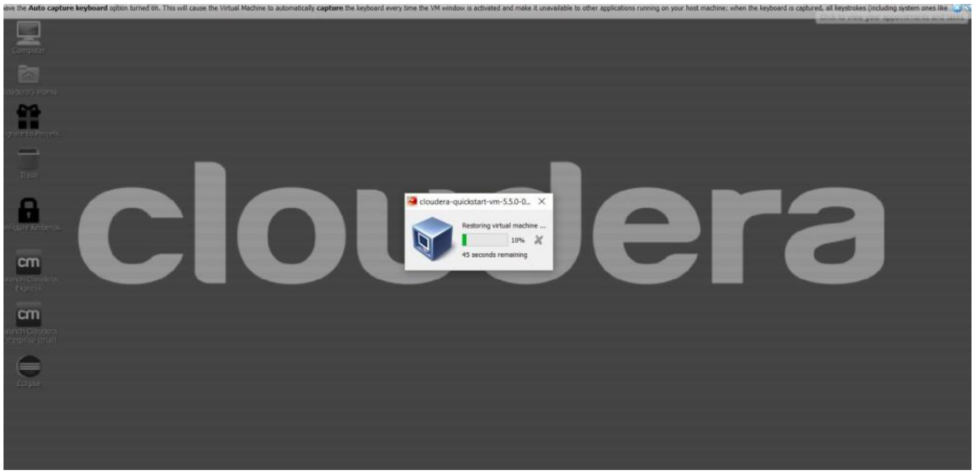
None

Start CDH



cloudera-quickstart-vm-5.5.0-0-virtualbox

Saved



ave the **Auto capture keyboard** option turned on. This will cause the Virtual Machine to automatically **capture** the keyboard every time the VM window is activated and make it unavailable to other applications running on your host machine: when the keyboard is captured, all keystrokes (including system ones like

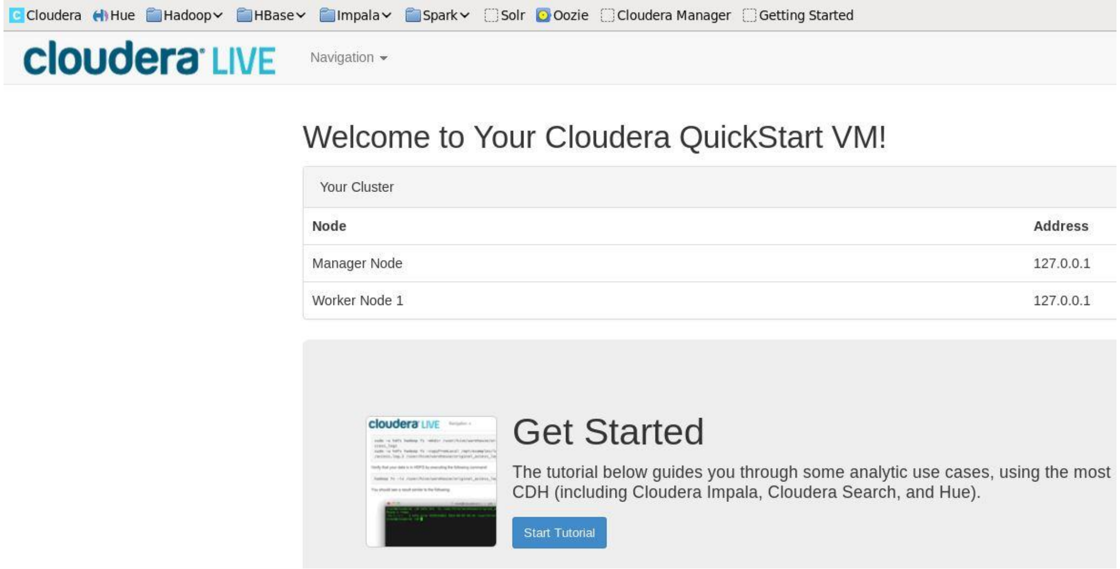
cloudera-quickstart-vm-5.5.0-0...

Restoring virtual machine ...

10%

45 seconds remaining

Use the Browser on the Host Machine to browse the Cloudera Quick Start Page



Welcome to Your Cloudera QuickStart VM!

Your Cluster

Node	Address
Manager Node	127.0.0.1
Worker Node 1	127.0.0.1

Get Started

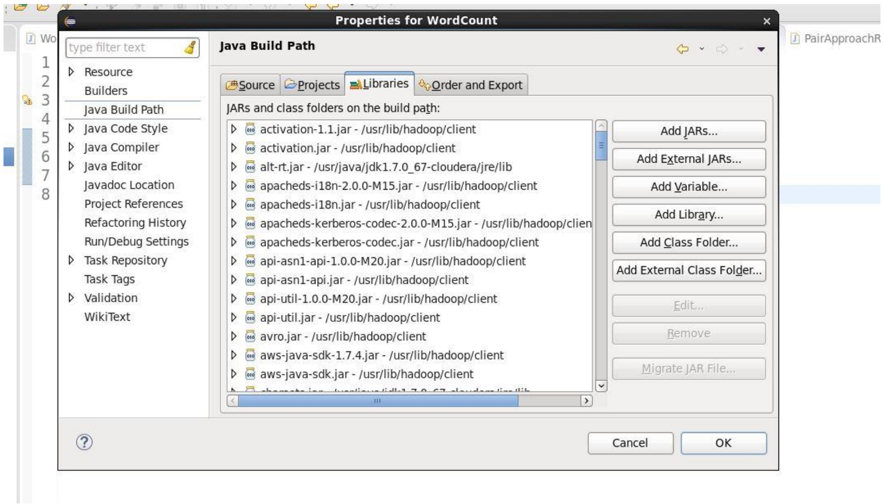
The tutorial below guides you through some analytic use cases, using the most CDH (including Cloudera Impala, Cloudera Search, and Hue).

Eclipse Project Setup

Create New Java Project

First, create a Java project in Eclipse inside the Host Machine (Linux). And, then add all the Hadoop related libraries as needed, for example –

/usr/lib/Hadoop/Hadoop.jar*
/usr/lib/Hadoop/client/.jar*

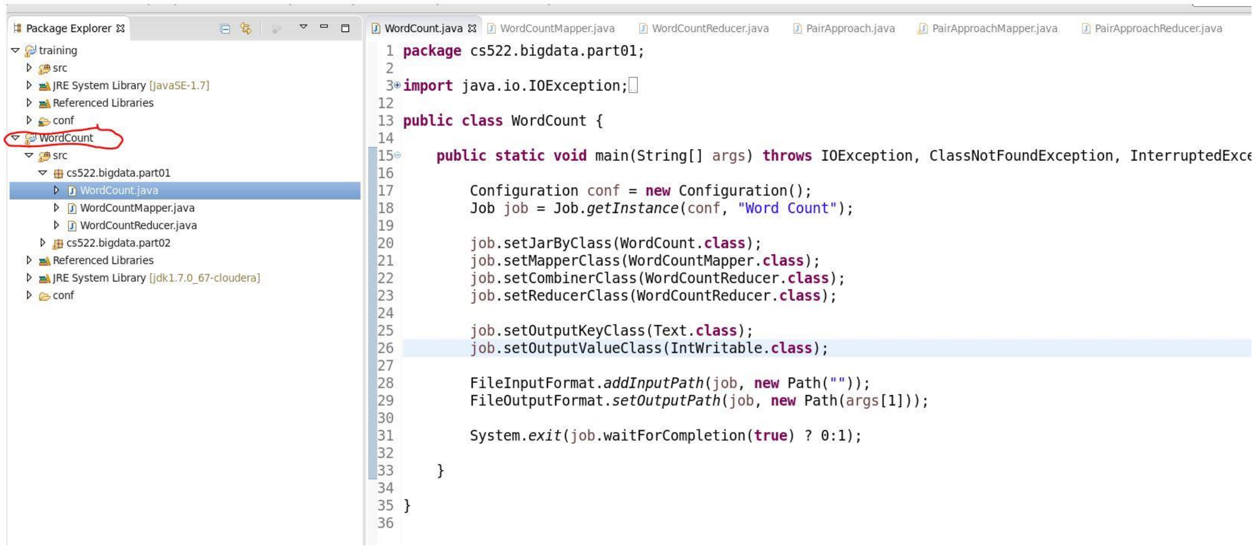


Java Build Path

JARs and class folders on the build path:

- activation-1.1.jar - /usr/lib/hadoop/client
- activation.jar - /usr/lib/hadoop/client
- alt-rt.jar - /usr/java/jdk1.7.0_67-cloudera/jre/lib
- apacheds-i18n-2.0.0-M15.jar - /usr/lib/hadoop/client
- apacheds-i18n.jar - /usr/lib/hadoop/client
- apacheds-kerberos-codec-2.0.0-M15.jar - /usr/lib/hadoop/clien
- apacheds-kerberos-codec.jar - /usr/lib/hadoop/client
- api-asn1-api-1.0.0-M20.jar - /usr/lib/hadoop/client
- api-asn1-api.jar - /usr/lib/hadoop/client
- api-util-1.0.0-M20.jar - /usr/lib/hadoop/client
- api-util.jar - /usr/lib/hadoop/client
- avro.jar - /usr/lib/hadoop/client
- aws-java-sdk-1.7.4.jar - /usr/lib/hadoop/client
- aws-java-sdk.jar - /usr/lib/hadoop/client

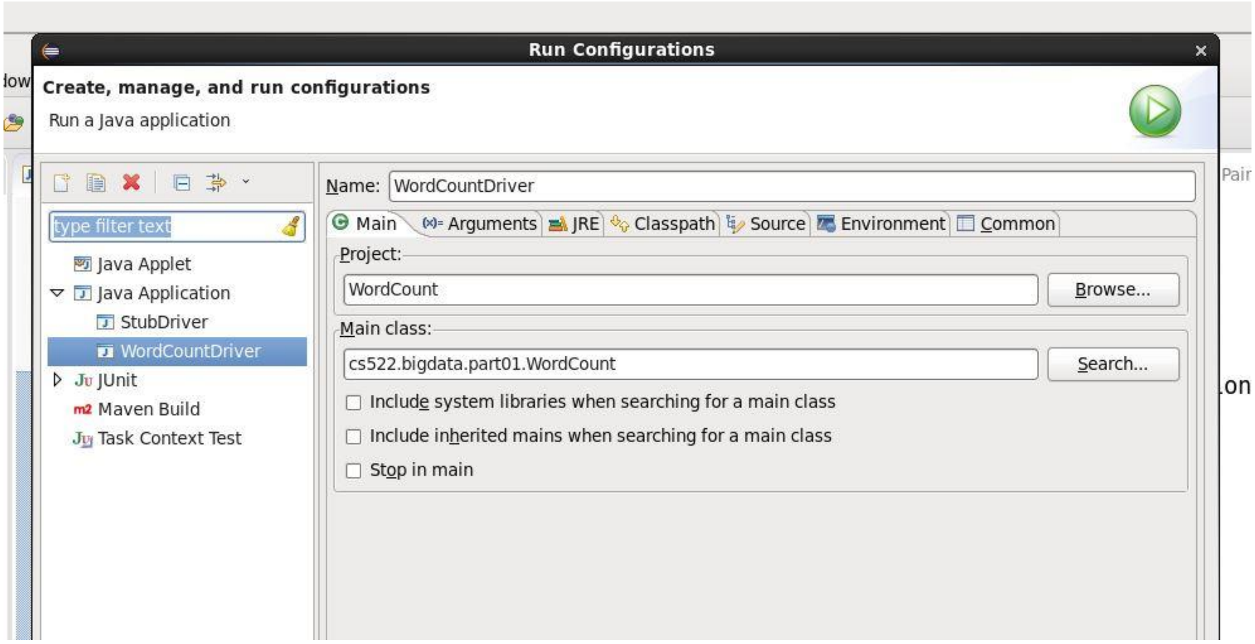
Write the WordCount Mapper and Reducer classes along with main() Method



```java
package cs522.bigdata.part01;

import java.io.IOException;

public class WordCount {

    public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedExce

        Configuration conf = new Configuration();
        Job job = Job.getInstance(conf, "Word Count");

        job.setJarByClass(WordCount.class);
        job.setMapperClass(WordCountMapper.class);
        job.setCombinerClass(WordCountReducer.class);
        job.setReducerClass(WordCountReducer.class);

        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);

        FileInputFormat.addInputPath(job, new Path(""));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));

        System.exit(job.waitForCompletion(true) ? 0:1);

    }

}
```

Export the Jar File

First, setup the Run Configurations as –


Run Configurations

Create, manage, and run configurations

Run a Java application

Name: WordCountDriver

Project: WordCount

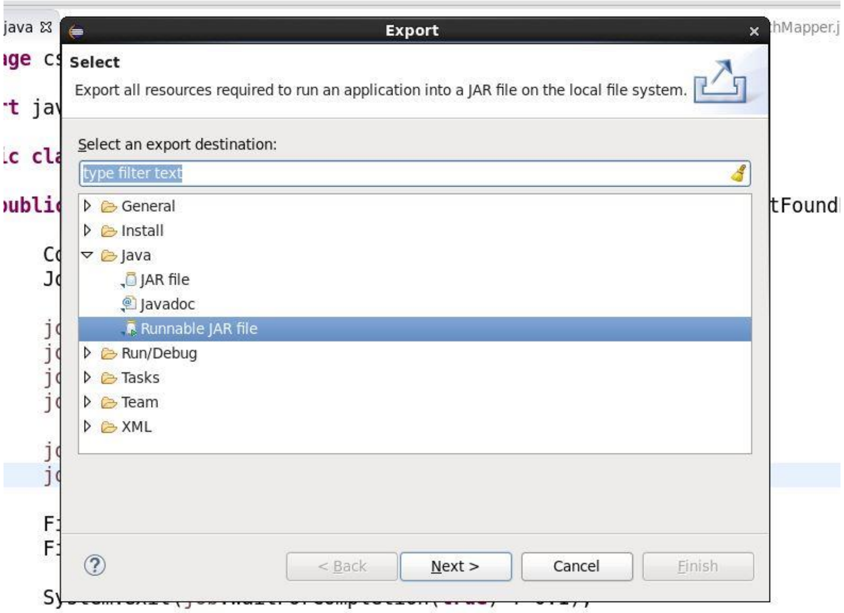
Main class: cs522.bigdata.part01.WordCount

□ Include system libraries when searching for a main class

□ Include inherited mains when searching for a main class

□ Stop in main

Right click on the project and click on Export –



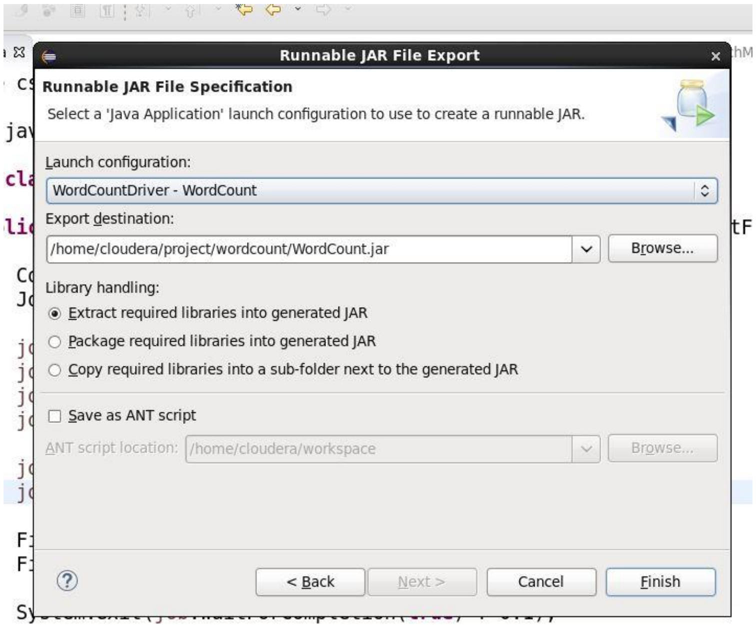
Export

Select

Export all resources required to run an application into a JAR file on the local file system.

Select an export destination:

type filter text



Runnable JAR File Export

Runnable JAR File Specification

Select a 'Java Application' launch configuration to use to create a runnable JAR.

Launch configuration: WordCountDriver - WordCount

Export destination: /home/cloudera/project/wordcount/WordCount.jar

Library handling:

● Extract required libraries into generated JAR

○ Package required libraries into generated JAR

○ Copy required libraries into a sub-folder next to the generated JAR

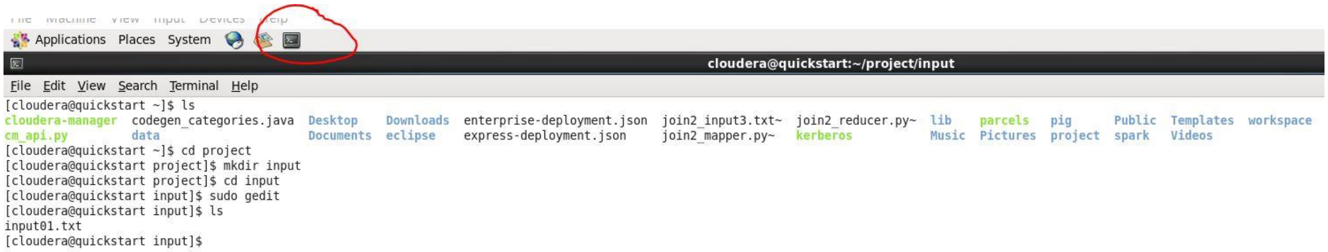
□ Save as ANT script

ANT script location: /home/cloudera/workspace

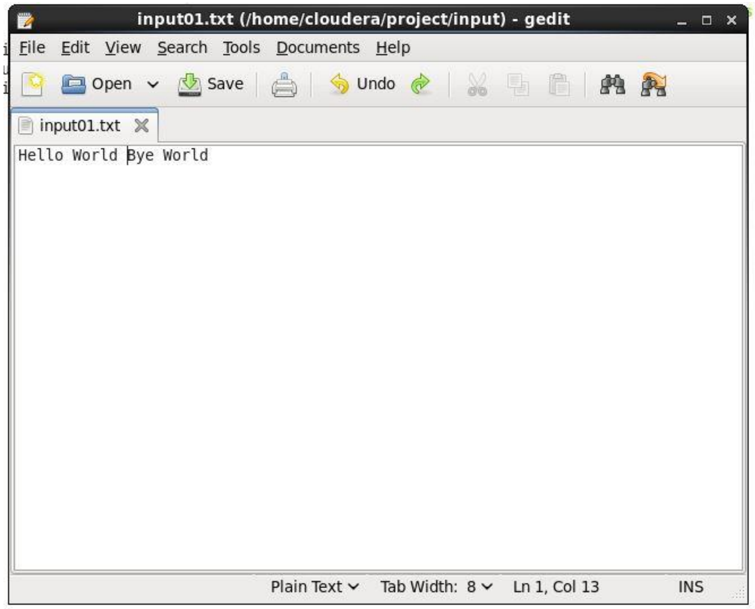
And, then click on finish button. It will finally export the jar file into the specified directory.

Run the Job

First, create an input file for the wordcount as follows –

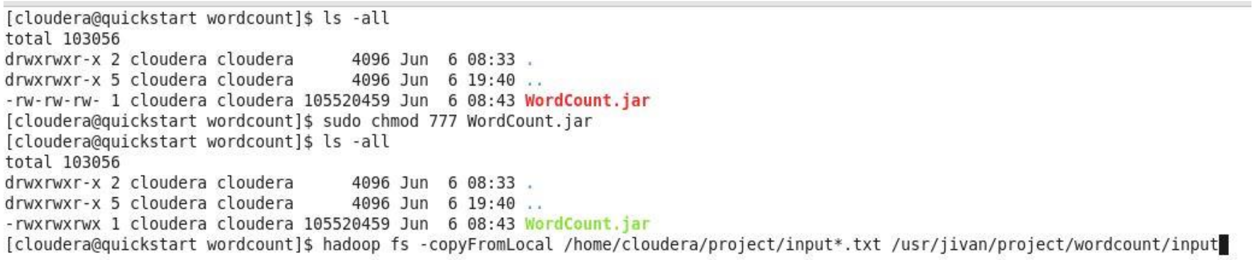


```
[cloudera@quickstart ~]$ ls
cloudera-manager  codegen_categories.java  Desktop    Downloads  enterprise-deployment.json  join2_input3.txt~  join2_reducer.py~  lib    parcels   pig      Public  Templates  workspace
cm_api.py         data                     Documents  eclipse    express-deployment.json     join2_mapper.py~   kerberos           Music  Pictures  project  spark   Videos
[cloudera@quickstart ~]$ cd project
[cloudera@quickstart project]$ mkdir input
[cloudera@quickstart project]$ cd input
[cloudera@quickstart input]$ sudo gedit
[cloudera@quickstart input]$ ls
input01.txt
[cloudera@quickstart input]$
```



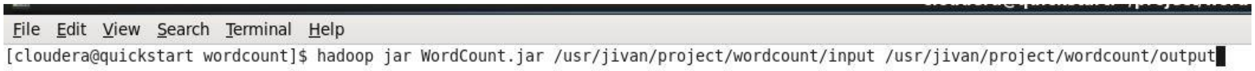
```
Hello World Bye World
```

Copy the file from local Linux FileSystem to the HDFS using the following commands –



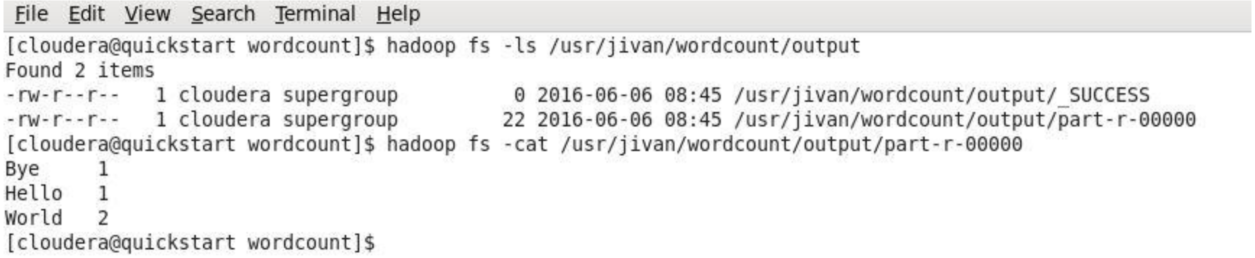
```
[cloudera@quickstart wordcount]$ ls -all
total 103056
drwxrwxr-x 2 cloudera cloudera      4096 Jun  6 08:33 .
drwxrwxr-x 5 cloudera cloudera      4096 Jun  6 19:40 ..
-rw-rw-rw- 1 cloudera cloudera 105520459 Jun  6 08:43 WordCount.jar
[cloudera@quickstart wordcount]$ sudo chmod 777 WordCount.jar
[cloudera@quickstart wordcount]$ ls -all
total 103056
drwxrwxr-x 2 cloudera cloudera      4096 Jun  6 08:33 .
drwxrwxr-x 5 cloudera cloudera      4096 Jun  6 19:40 ..
-rwxrwxrwx 1 cloudera cloudera 105520459 Jun  6 08:43 WordCount.jar
[cloudera@quickstart wordcount]$ hadoop fs -copyFromLocal /home/cloudera/project/input*.txt /usr/jivan/project/wordcount/input
```

Now, run the Jar file as follows, make sure the output directory does not exists as it will be created during execution –



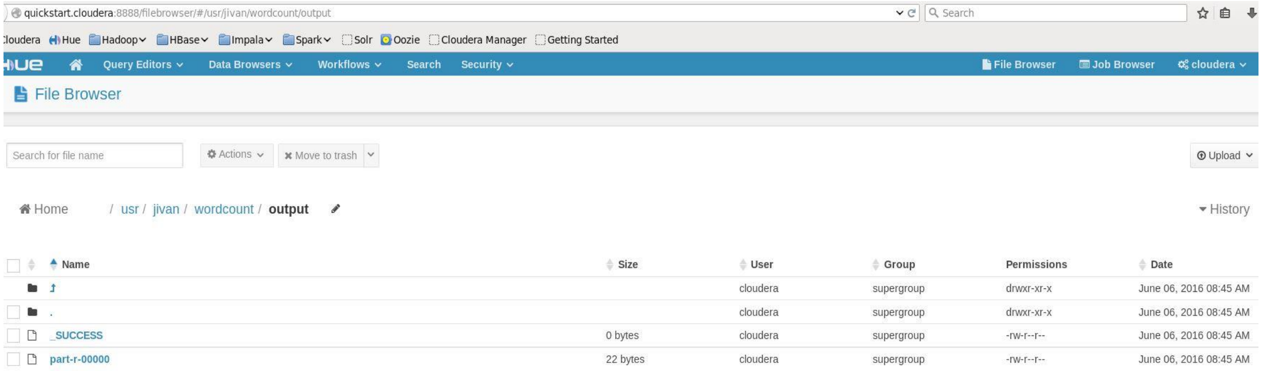
```
[cloudera@quickstart wordcount]$ hadoop jar WordCount.jar /usr/jivan/project/wordcount/input /usr/jivan/project/wordcount/output
```

This job execution takes a little bit time and when the map task and reduce task complete to 100%, then check for the output file in the output directory as specified in the above command –



```
[cloudera@quickstart wordcount]$ hadoop fs -ls /usr/jivan/wordcount/output
Found 2 items
-rw-r--r--   1 cloudera supergroup          0 2016-06-06 08:45 /usr/jivan/wordcount/output/_SUCCESS
-rw-r--r--   1 cloudera supergroup         22 2016-06-06 08:45 /usr/jivan/wordcount/output/part-r-00000
[cloudera@quickstart wordcount]$ hadoop fs -cat /usr/jivan/wordcount/output/part-r-00000
Bye     1
Hello   1
World   2
[cloudera@quickstart wordcount]$
```

And, it verifies the word count result! We can upload the input/check output files to/in HDFS using the Hue Interface from the browser as follows –



	Name	Size	User	Group	Permissions	Date
	⮥		cloudera	supergroup	drwxr-xr-x	June 06, 2016 08:45 AM
□	.		cloudera	supergroup	drwxr-xr-x	June 06, 2016 08:45 AM
□	_SUCCESS	0 bytes	cloudera	supergroup	-rw-r--r--	June 06, 2016 08:45 AM
□	part-r-00000	22 bytes	cloudera	supergroup	-rw-r--r--	June 06, 2016 08:45 AM

(The default credentials are – user: cloudera, pass: cloudera)

Next – we‘ll be working on the Pair Approach implementation for finding the relative frequencies.