**Installing Hadoop 3.2.1 Single node cluster on Windows 10**

<https://towardsdatascience.com/installing-hadoop-3-2-1-single-node-cluster-on-windows-10-ac258dd48aef>

While working on a [project](https://www.researchgate.net/project/ORADIEX-A-Big-Data-driven-smart-framework-for-real-time-surveillance-and-analysis-of-individual-exposure-to-radioactive-pollution) two years ago, I wrote a step-by-step guide to [install Hadoop 3.1.0 on Ubuntu 16.04](https://towardsdatascience.com/installing-hadoop-3-1-0-multi-node-cluster-on-ubuntu-16-04-step-by-step-8d1954b31505) operating system. Since we are currently working on a new project where we need to install a Hadoop cluster on Windows 10, I decided to write a guide for this process.

This article is a part of a series that we are publishing on TowardsDataScience.com that aims to illustrate how to install Big Data technologies on Windows operating system.

**1. Prerequisites**

First, we need to make sure that the following prerequisites are installed:

1. Java 8 runtime environment (JRE): [Hadoop 3 requires a Java 8 installation](https://cwiki.apache.org/confluence/display/HADOOP/Hadoop+Java+Versions). I prefer using the [offline installer](https://www.java.com/en/download/windows_offline.jsp).

2. [Java 8 development Kit (JDK)](https://www.oracle.com/java/technologies/javase-jdk8-downloads.html)

<https://www.oracle.com/java/technologies/downloads/#java8-windows>

download windows – Java 8

|  |  |  |
| --- | --- | --- |
| x64 Installer | 145.55 MB | [jdk-8u381-windows-x64.exe](https://www.oracle.com/java/technologies/downloads/#license-lightbox) |

3. To unzip downloaded Hadoop binaries, we should install [7zip](https://www.7-zip.org/download.html).

4. I will create a folder “E:\hadoop-env” on my local machine to store downloaded files.

**2. Download Hadoop binaries**

The first step is to download Hadoop binaries from the [official website](https://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-3.2.1/hadoop-3.2.1.tar.gz). The binary package size is about 342 MB.

Or

<https://hadoop.apache.org/release/3.2.1.html>



Figure 1 — Hadoop binaries download link

After finishing the file download, we should unpack the package using 7zip int two steps. First, we should extract the hadoop-3.2.1.tar.gz library, and then, we should unpack the extracted tar file:

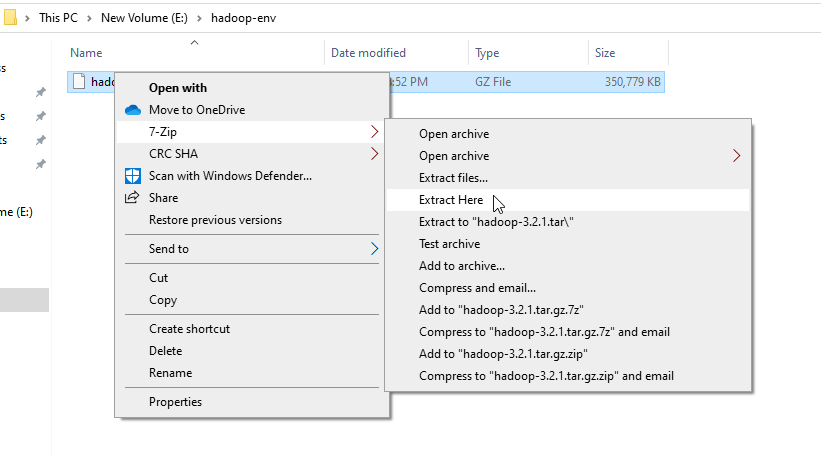


Figure 2 — Extracting hadoop-3.2.1.tar.gz package using 7zip

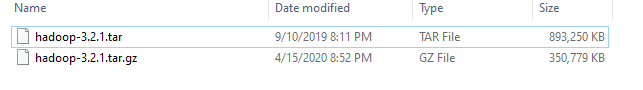


Figure 3 — Extracted hadoop-3.2.1.tar file

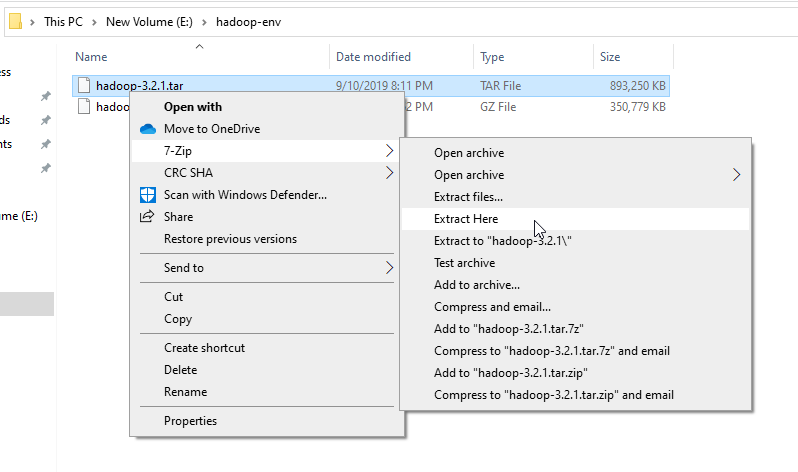


Figure 4 — Extracting the hadoop-3.2.1.tar file

The tar file extraction may take some minutes to finish. In the end, you may see some warnings about symbolic link creation. Just ignore these warnings since they are not related to windows.

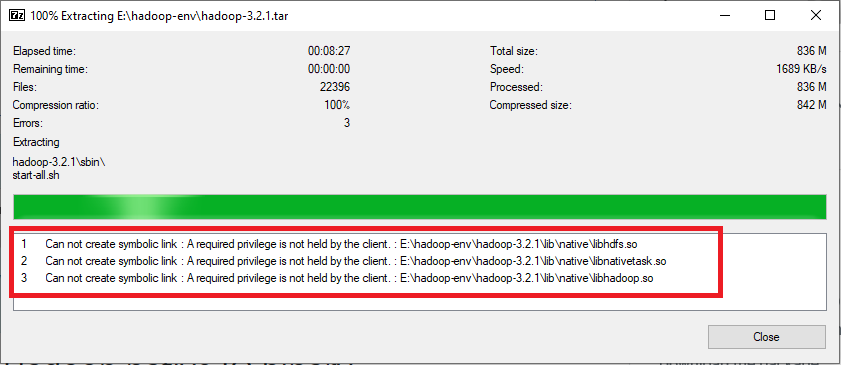


Figure 5 — Symbolic link warnings

After unpacking the package, we should add the Hadoop native IO libraries, which can be found in the following GitHub repository: <https://github.com/cdarlint/winutils>.

Since we are installing Hadoop 3.2.1, we should download the files located in <https://github.com/cdarlint/winutils/tree/master/hadoop-3.2.1/bin> and copy them into the “hadoop-3.2.1\bin” directory.

Download the full master zip file of winutils and copy the contents of bin folder of Hadoop-3.2.1

<https://github.com/cdarlint/winutils/tree/master>

**3. Setting up environment variables**

After installing Hadoop and its prerequisites, we should configure the environment variables to define Hadoop and Java default paths.

To edit environment variables, go to Control Panel > System and Security > System (or right-click > properties on My Computer icon) and click on the “Advanced system settings” link.

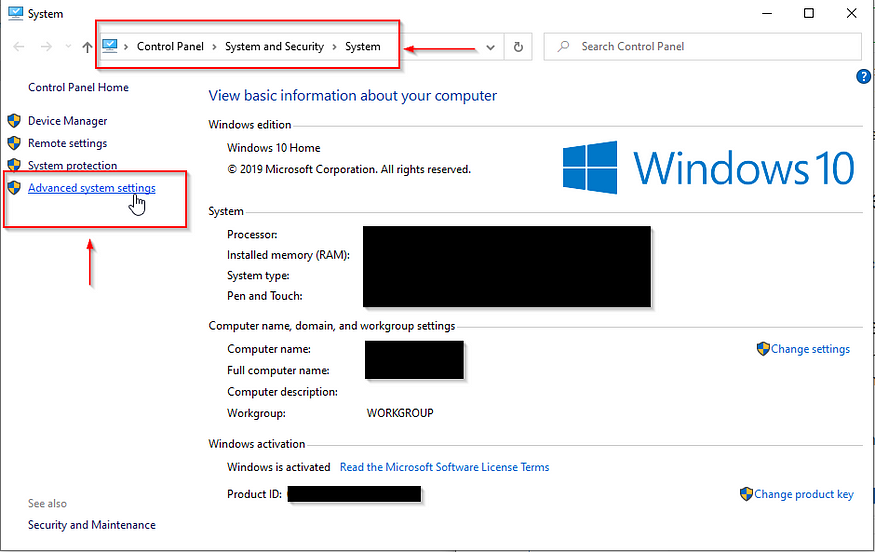


Figure 6 — Opening advanced system settings

When the “Advanced system settings” dialog appears, go to the “Advanced” tab and click on the “Environment variables” button located on the bottom of the dialog.

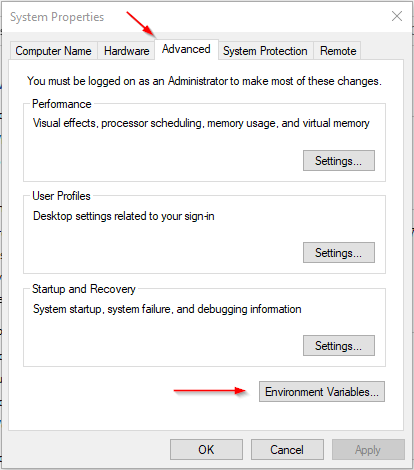


Figure 7 — Advanced system settings dialog

In the “Environment Variables” dialog, press the “New” button to add a new variable.

*Note: In this guide, we will add* ***user variables since we are configuring Hadoop for a single user.*** *If you are looking to configure Hadoop for multiple users, you can define System variables instead.*

There are two variables to define:

1. JAVA\_HOME: JDK installation folder path

2. HADOOP\_HOME: Hadoop installation folder path

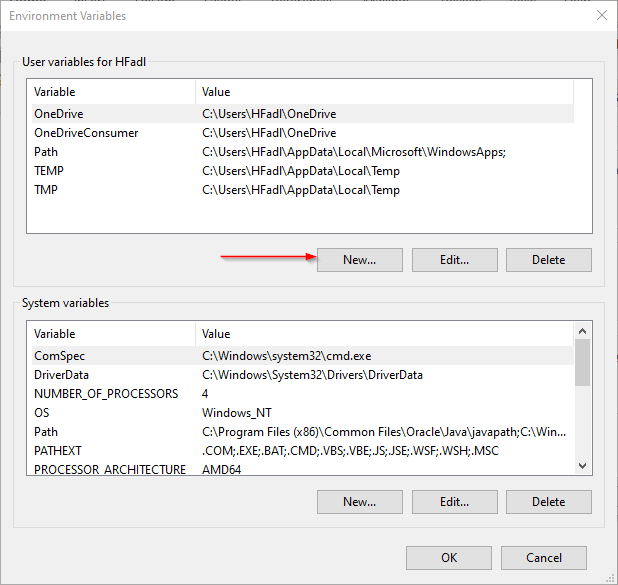


Figure 8 — Adding JAVA\_HOME variable

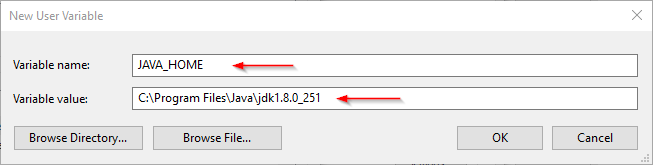


Figure 9 — Adding HADOOP\_HOME variable

Now, we should edit the PATH variable to add the Java and Hadoop binaries paths as shown in the following screenshots.

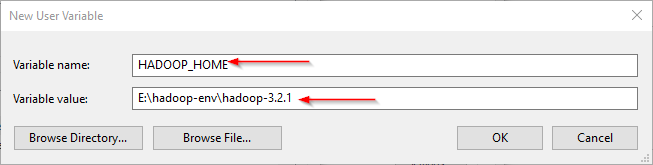


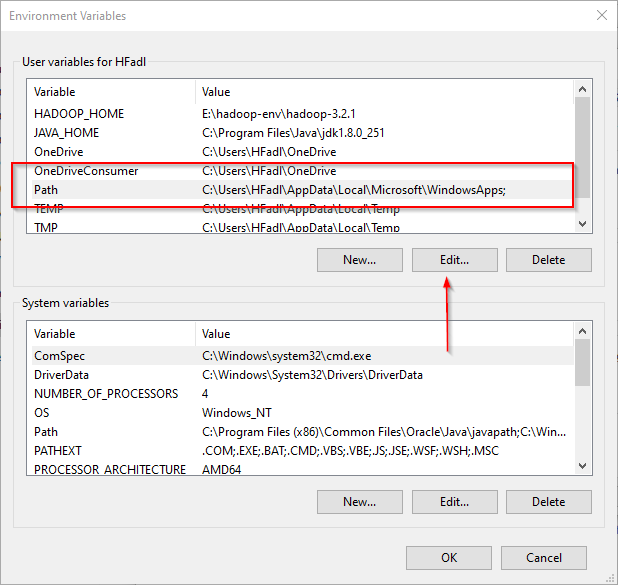
Figure 10 — Editing the PATH variable

Figure 11 — Editing PATH variable

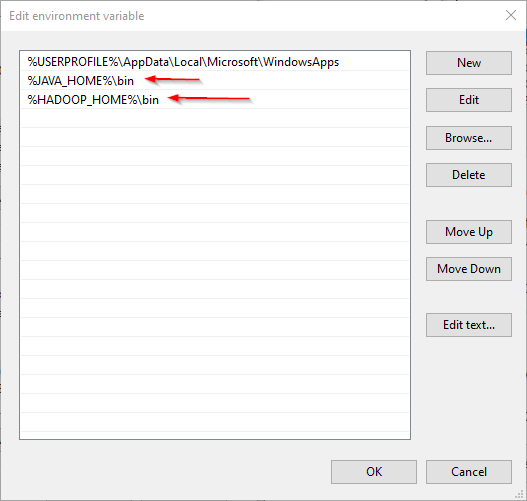


Figure 12— Adding new paths to the PATH variable

**3.1. JAVA\_HOME is incorrectly set error**

Now, let’s open PowerShell and try to run the following command:

hadoop -version

In this example, since the JAVA\_HOME path contains spaces, I received the following error:

JAVA\_HOME is incorrectly set

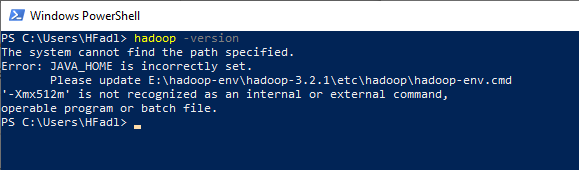


Figure 13 — JAVA\_HOME error

To solve this issue, we should use the windows 8.3 path instead. As an example:

* Use “Progra~1” instead of “Program Files”
* Use “Progra~2” instead of “Program Files(x86)”

After replacing “Program Files” with “Progra~1”, we closed and reopened PowerShell and tried the same command. As shown in the screenshot below, it runs without errors.

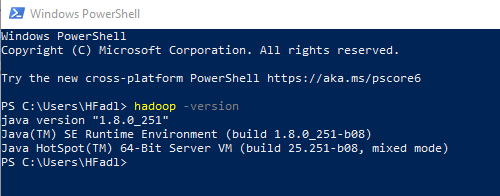


Figure 14 — hadoop -version command executed successfully

**4. Configuring Hadoop cluster**

There are four files we should alter to configure Hadoop cluster:

1. %HADOOP\_HOME%\etc\hadoop\hdfs-site.xml
2. %HADOOP\_HOME%\etc\hadoop\core-site.xml
3. %HADOOP\_HOME%\etc\hadoop\mapred-site.xml
4. %HADOOP\_HOME%\etc\hadoop\yarn-site.xml

**4.1. HDFS site configuration**

As we know, Hadoop is built using a master-slave paradigm. Before altering the HDFS configuration file, we should create a directory to store all master node (name node) data and another one to store data (data node). In this example, we created the following directories:

* E:\hadoop-env\hadoop-3.2.1\data\dfs\namenode
* E:\hadoop-env\hadoop-3.2.1\data\dfs\datanode

Now, let’s open “hdfs-site.xml” file located in “%HADOOP\_HOME%\etc\hadoop” directory, and we should add the following properties within the <configuration></configuration> element:

<property><name>dfs.replication</name><value>1</value></property><property><name>dfs.namenode.name.dir</name><value>file:///E:/hadoop-env/hadoop-3.2.1/data/dfs/namenode</value></property><property><name>dfs.datanode.data.dir</name><value>file:///E:/hadoop-env/hadoop-3.2.1/data/dfs/datanode</value></property>

*Note that we have set the replication factor to 1 since we are creating a single node cluster.*

**4.2. Core site configuration**

Now, we should configure the name node URL adding the following XML code into the <configuration></configuration> element within “core-site.xml”:

<property><name>fs.default.name</name><value>hdfs://localhost:9820</value></property>

**4.3. Map Reduce site configuration**

Now, we should add the following XML code into the <configuration></configuration> element within “mapred-site.xml”:

<property><name>mapreduce.framework.name</name><value>yarn</value><description>MapReduce framework name</description></property>

**4.4. Yarn site configuration**

Now, we should add the following XML code into the <configuration></configuration> element within “yarn-site.xml”:

<property><name>yarn.nodemanager.aux-services</name><value>mapreduce\_shuffle</value><description>Yarn Node Manager Aux Service</description></property>

**5. Formatting Name node**

After finishing the configuration, let’s try to format the name node using the following command:

hdfsnamenode -format

Due to a [bug in the Hadoop 3.2.1 release](https://issues.apache.org/jira/browse/HDFS-14890), you will receive the following error:

2020–04–17 22:04:01,503 ERROR namenode.NameNode: Failed to start namenode.java.lang.UnsupportedOperationExceptionatjava.nio.file.Files.setPosixFilePermissions(Files.java:2044)at org.apache.hadoop.hdfs.server.common.Storage$StorageDirectory.clearDirectory(Storage.java:452)at org.apache.hadoop.hdfs.server.namenode.NNStorage.format(NNStorage.java:591)at org.apache.hadoop.hdfs.server.namenode.NNStorage.format(NNStorage.java:613)at org.apache.hadoop.hdfs.server.namenode.FSImage.format(FSImage.java:188)at org.apache.hadoop.hdfs.server.namenode.NameNode.format(NameNode.java:1206)at org.apache.hadoop.hdfs.server.namenode.NameNode.createNameNode(NameNode.java:1649)at org.apache.hadoop.hdfs.server.namenode.NameNode.main(NameNode.java:1759)2020–04–17 22:04:01,511 INFO util.ExitUtil: Exiting with status 1: java.lang.UnsupportedOperationException2020–04–17 22:04:01,518 INFO namenode.NameNode: SHUTDOWN\_MSG:

This issue will be solved within the next release. For now, you can fix it temporarily using the following steps ([reference](https://kontext.tech/column/hadoop/377/latest-hadoop-321-installation-on-windows-10-step-by-step-guide)):

1. Download hadoop-hdfs-3.2.1.jar file from the [following link](https://github.com/FahaoTang/big-data/blob/master/hadoop-hdfs-3.2.1.jar).

<https://github.com/FahaoTang/big-data/blob/master/hadoop-hdfs-3.2.1.jar>

download the raw file on right size Icon --- size should be > 5MB

1. Rename the file name hadoop-hdfs-3.2.1.jar (already available in this folder )to hadoop-hdfs-3.2.1.bak in folder %HADOOP\_HOME%\share\hadoop\hdfs
2. Copy the downloaded hadoop-hdfs-3.2.1.jar to folder %HADOOP\_HOME%\share\hadoop\hdfs

Now, if we try to re-execute the format command (Run the command prompt or PowerShell as administrator), you need to approve file system format.

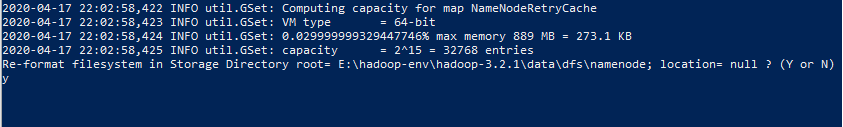


Figure 15 — File system format approval

And the command is executed successfully:

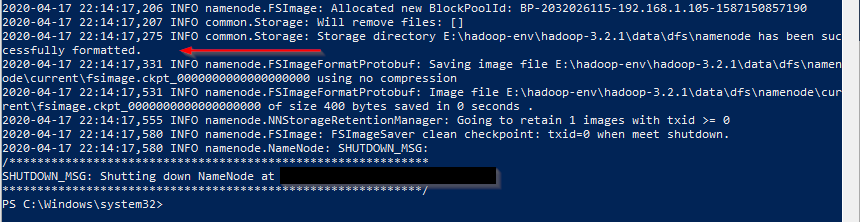


Figure 16 — Command executed successfully

**6. Starting Hadoop services**

Now, we will open PowerShell, and navigate to “%HADOOP\_HOME%\sbin” directory. Then we will run the following command to start the Hadoop nodes:

.\start-dfs.cmd



Figure 17 — StartingHadoop nodes

Two command prompt windows will open (one for the name node and one for the data node) as follows:

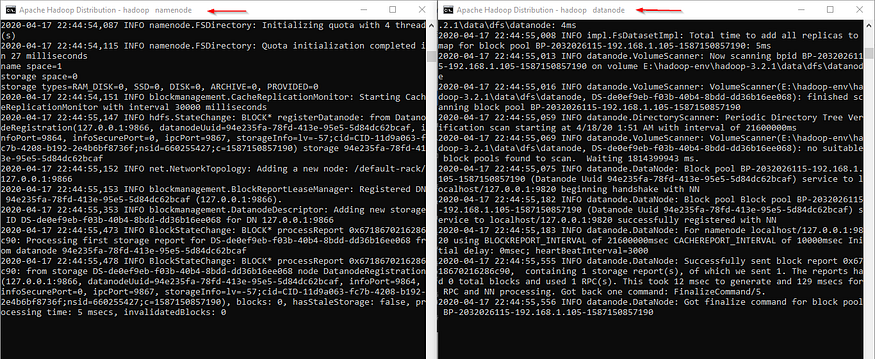


Figure 18 — Hadoop nodes command prompt windows

Next, we must start the Hadoop Yarn service using the following command:

./start-yarn.cmd



Figure 19 — Starting Hadoop Yarn services

Two command prompt windows will open (one for the resource manager and one for the node manager) as follows:

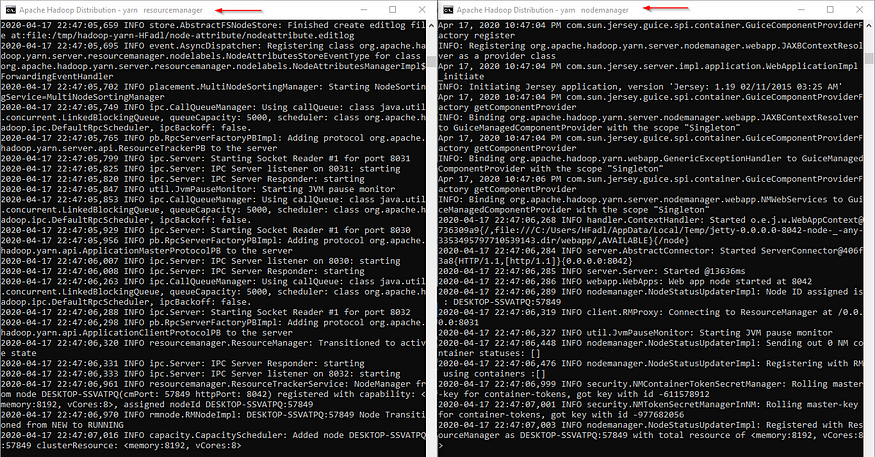


Figure 20— Node manager and Resource manager command prompt windows

To make sure that all services started successfully, we can run the following command:

jps

It should display the following services:

14560 DataNode  
4960 ResourceManager  
5936 NameNode  
768 NodeManager  
14636 Jps



Figure 21 — Executing jps command

**7. Hadoop Web UI**

There are three web user interfaces to be used:

* Name node web page: <http://localhost:9870/dfshealth.html>

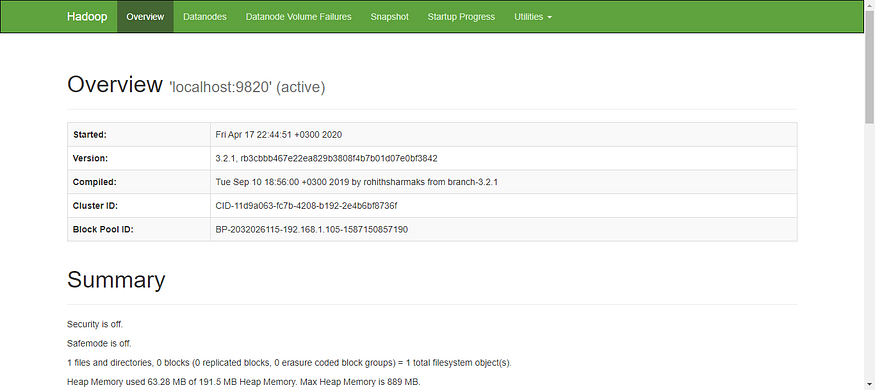


Figure 22 — Name node web page

* Data node web page: <http://localhost:9864/datanode.html>

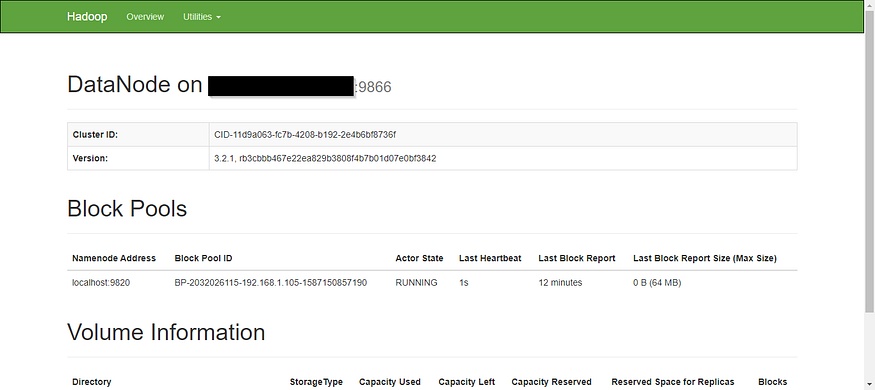


Figure 23 — Data node web page

* Yarn web page: <http://localhost:8088/cluster>

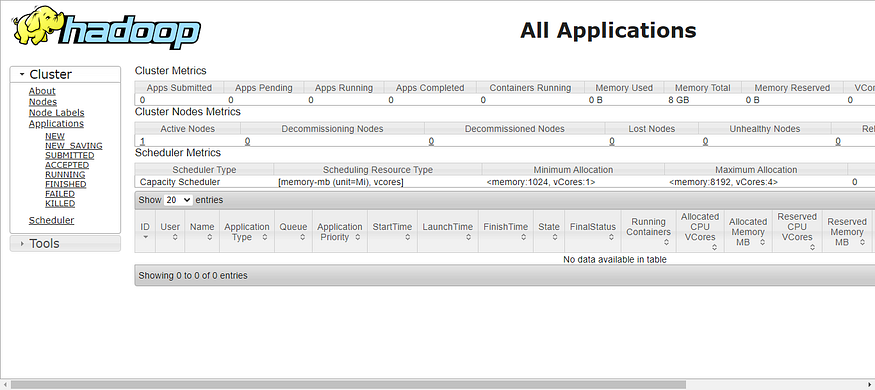


Figure 24 — Yarn web page

[**https://medium.com/mcd-unison/installing-hadoop-3-2-1-in-windows-10-basic-text-analysis-4a5717c209ae**](https://medium.com/mcd-unison/installing-hadoop-3-2-1-in-windows-10-basic-text-analysis-4a5717c209ae)

NOTE: If after you end services and start again, the command prompt shows a problem on the \temp folder, open a command prompt as administrator and run the following command: (The problem is due to permissions of the Hadoop folders)

cacls hadoop-3.2.1 /t /p everyone:f

# Basic word count example using Hadoop’s Mapreduce

In this section, we’ll analyze the most common words in the novel “[Don Quijote de la Mancha](https://es.wikipedia.org/wiki/Don_Quijote_de_la_Mancha)” by [Miguel de Cervantes Saavedra](https://es.wikipedia.org/wiki/Miguel_de_Cervantes).

For this purpose, we’ll need to download the novel as a plain text file, which you can find on this link: <https://gist.githubusercontent.com/jsdario/6d6c69398cb0c73111e49f1218960f79/raw/8d4fc4548d437e2a7203a5aeeace5477f598827d/el_quijote.txt>

Once downloaded, move the file to “C:”.

Open a command prompt as administrator and run the following command to create an input and output folder on the Hadoop file system, to which we will be moving the novel for our analysis.

hadoop fs -mkdir /input\_dir  
hadoop fs -mkdir /output\_dir

We can verify the folder we just created on the following link: <http://localhost:9870/explorer.html#/>

Now let’s move the file “el\_quijote.txt”, from C: to /input\_dir

hadoop fs -put C:/el\_quijote.txt /input\_dir

We can verify the file has moved successfully on the following link:

<http://localhost:9870/explorer.html#/input_dir>

or with the following command line:

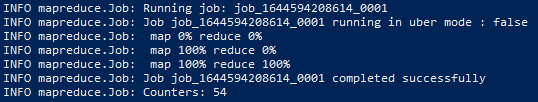
hadoop fs -ls /input\_dir

For the next step, we’ll need the “hadoop-mapreduce-examples-3.2.1.jar”, found in “C:\hadoop-3.2.1\share\hadoop\mapreduce”.

We’ll execute the following command in order to call the function wordcount inside the mapreduce.

hadoop jar C:\hadoop-3.2.1\share\hadoop\mapreduce\hadoop-mapreduce-examples-3.2.1.jar wordcount /input\_dir/el\_quijote.txt /output\_dir

Once the script is executed, the following message will appear on the command prompt



Command prompt message

Now you can find the results of the wordcount on the output\_dir we created.

Open up the output\_dir folder inside the Hadoop interface on the following link: <http://localhost:9870/explorer.html#/output_dir> and click the file named “part-r-00000”**(1)**, and a small window will show up, click the “download”**(2)** option, and there you will be able to see the number of appearances for each word in the novel.

**Now we are done!**

**Copy the file from hadoop to local drive**

hadoop fs -get /output\_dir/part-r-00000.txt C:/output.txt