## Introduction

Hadoop is a software framework from Apache Software Foundation that is used to store and process Big Data. It has two main components; Hadoop Distributed File System (HDFS), its storage system and MapReduce, is its data processing framework. Hadoop has the capability to manage large datasets by distributing the dataset into smaller chunks across multiple machines and performing parallel computation on it .
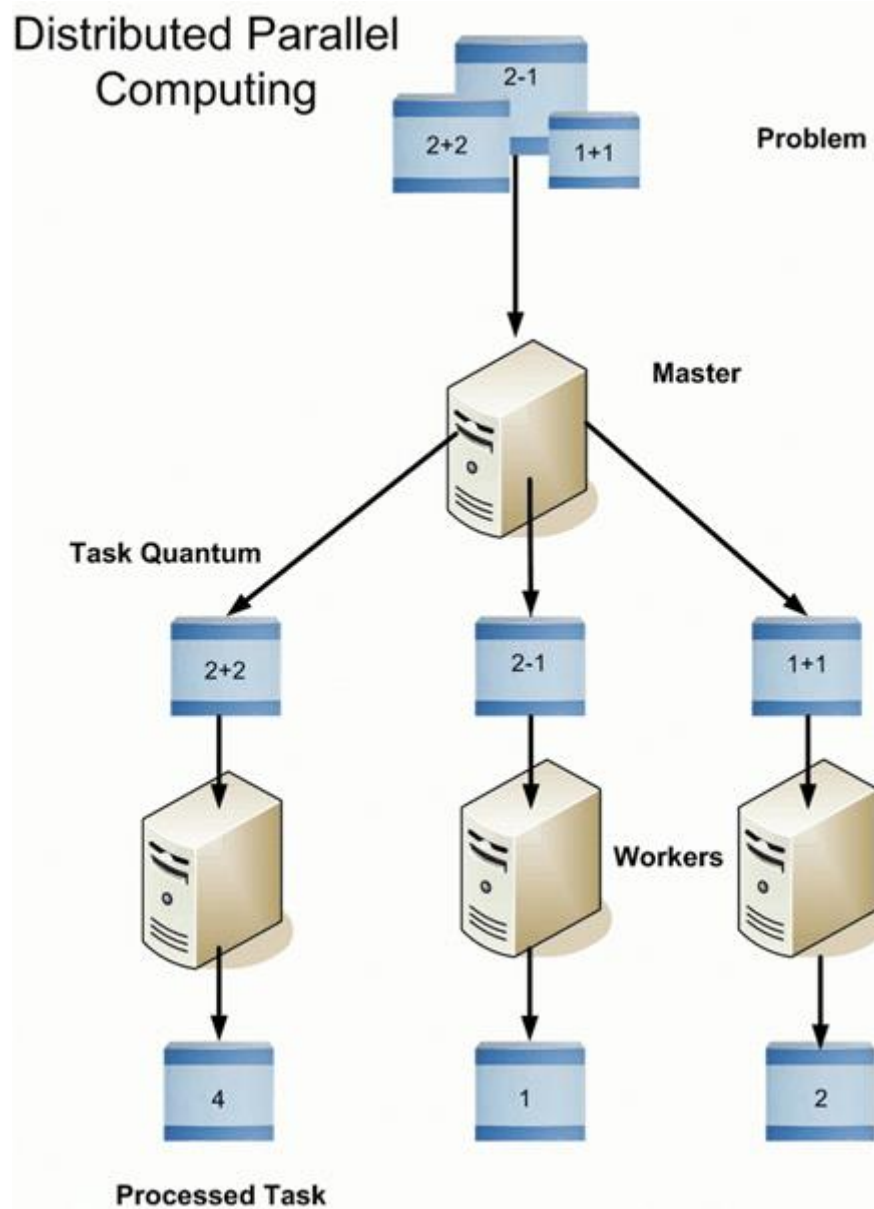

Overview of HDFS

Hadoop is an essential component of the Big Data industry as it provides the most reliable storage layer, HDFS, which can scale massively. Companies like Yahoo and Facebook use HDFS to store their data.

HDFS has a master-slave architecture where the master node is called NameNode and slave node is called DataNode. The NameNode and its DataNodes form a cluster. NameNode acts like an instructor to DataNode while the DataNodes store the actual data.
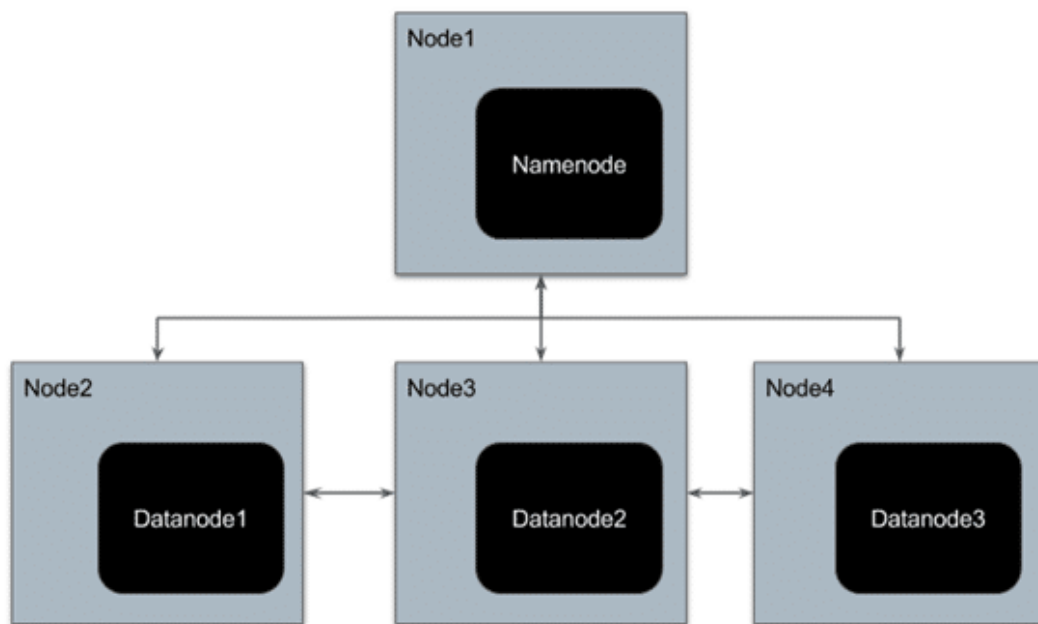
source: Hasura

There is another component of Hadoop known as YARN. The idea of Yarn is to manage the resources and schedule/monitor jobs in Hadoop. Yarn has two main components, Resource Manager and Node Manager. The resource manager has the authority to allocate resources to various applications running in a cluster. The node manager is responsible for monitoring their resource usage (CPU, memory, disk) and reporting the same to the resource manager.

Hadoop Yarn architecture
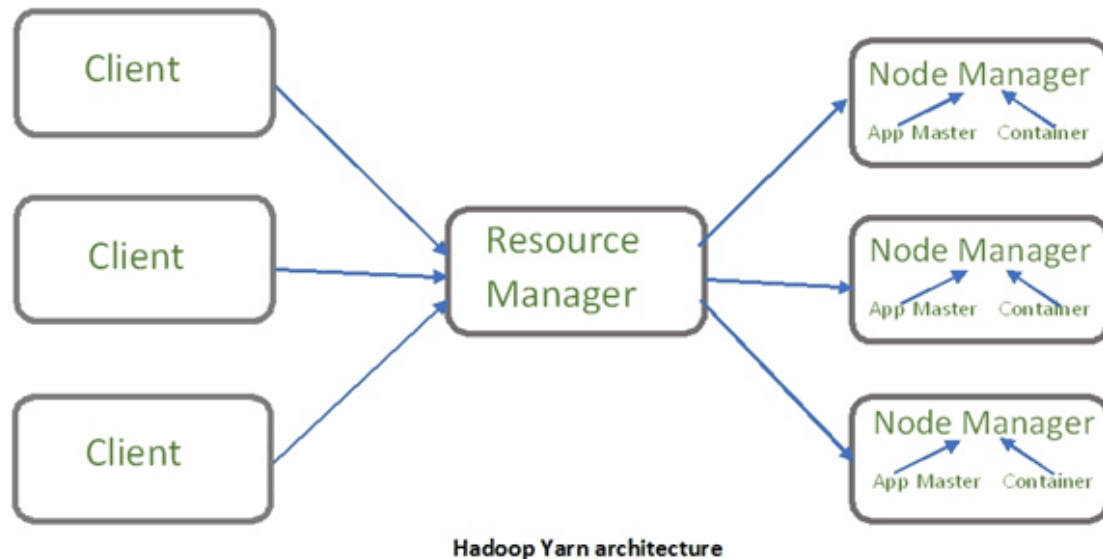
source: GeeksforGeeks

To understand the Hadoop architecture in detail, refer this blog –

Advantages of Hadoop

1. Economical – Hadoop is an open source Apache product, so it is free software. It has hardware cost associated with it. It is cost effective as it uses commodity hardware that are cheap machines to store its datasets and not any specialized machine.

2. Scalable – Hadoop distributes large data sets across multiple machines of a cluster. New machines can be easily added to the nodes of a cluster and can scale to thousands of nodes storing thousands of terabytes of data.

3. Fault Tolerance – Hadoop, by default, stores 3 replicas of data across the nodes of a cluster. So if any node goes down, data can be retrieved from other nodes.

4. Fast – Since Hadoop processes distributed data parallelly, it can process large data sets much faster than the traditional systems. It is highly suitable for batch processing of data.

5. Flexibility – Hadoop can store structured, semi-structured as well as unstructured data. It can accept data in the form of textfile, images, CSV files, XML files, emails, etc

6. Data Locality – Traditionally, to process the data, the data was fetched from the location it is stored, to the location where the application is submitted; however, in Hadoop, the processing application goes to the location of data to perform computation. This reduces the delay in processing of data.

7. Compatibility – Most of the emerging big data tools can be easily integrated with Hadoop like Spark. They use Hadoop as a storage platform and work as its processing system.

Hadoop Deployment Methods

1. Standalone Mode – It is the default mode of configuration of Hadoop. It doesn't use hdfs instead, it uses a local file system for both input and output. It is useful for debugging and testing.

2. Pseudo-Distributed Mode – It is also called a single node cluster where both NameNode and DataNode resides in the same machine. All the daemons run on the same machine in this mode. It produces a fully functioning cluster on a single machine.

3. Fully Distributed Mode – Hadoop runs on multiple nodes wherein there are separate nodes for master and slave daemons. The data is distributed among a cluster of machines providing a production environment.

Hadoop Installation on Windows 10

As a beginner, you might feel reluctant in performing cloud computing which requires subscriptions. While you can install a virtual machine as well in your system, it requires allocation of a large amount of RAM for it to function smoothly else it would hang constantly.

You can install Hadoop in your system as well which would be a feasible way to learn Hadoop.

We will be installing single node pseudo-distributed hadoop cluster on windows 10.

Prerequisite: To install Hadoop, you should have Java version 1.8 in your system.

Check your java version through this command on command prompt

1 java –version

If java is not installed in your system, then –

Go this link –

https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downl...

Accept the license,

Download the file according to your operating system. Keep the java folder directly under the local disk directory (C:\Java\jdk1.8.0_152) rather than in Program Files (C:\Program Files\Java\jdk1.8.0_152) as it can create errors afterwards.

This PC > Local Disk (C:) > Java		
Name	Date modified	Type
jdk1.8.0_152	4/7/2019 2:54 PM	File folder

After downloading java version 1.8, download hadoop version 3.1 from this link –

https://archive.apache.org/dist/hadoop/common/hadoop-3.1.0/hadoop-3...

Extract it to a folder.

; PC > Downloads > hadoop-3.1.0 > hadoop-3.1.0			
Name	Date modified	Type	Size
bin	4/7/2019 8:24 PM	File folder	
etc	4/7/2019 8:24 PM	File folder	
include	4/7/2019 8:24 PM	File folder	
lib	4/7/2019 8:24 PM	File folder	
libexec	4/7/2019 8:24 PM	File folder	
sbin	4/7/2019 8:24 PM	File folder	
share	4/7/2019 8:16 PM	File folder	
LICENSE	3/21/2018 11:27 PM	Text Document	144 KB
NOTICE	3/21/2018 11:27 PM	Text Document	22 KB
README	3/21/2018 11:27 PM	Text Document	2 KB

Setup System Environment Variables

Open control panel to edit the system environment variable

← → ˅ ↑ 🖳 > Control Panel

System
Edit environment variables for your account
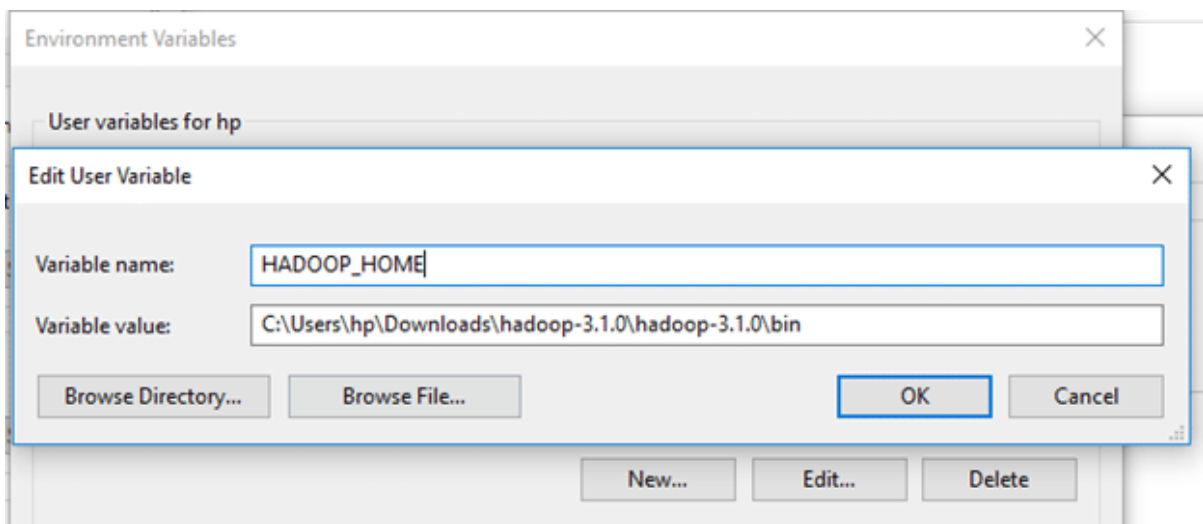🛡 Edit the system environment variables
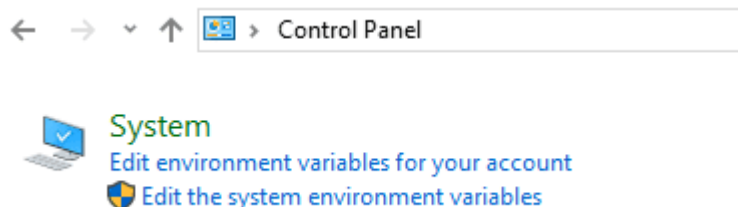
Go to environment variable in system properties
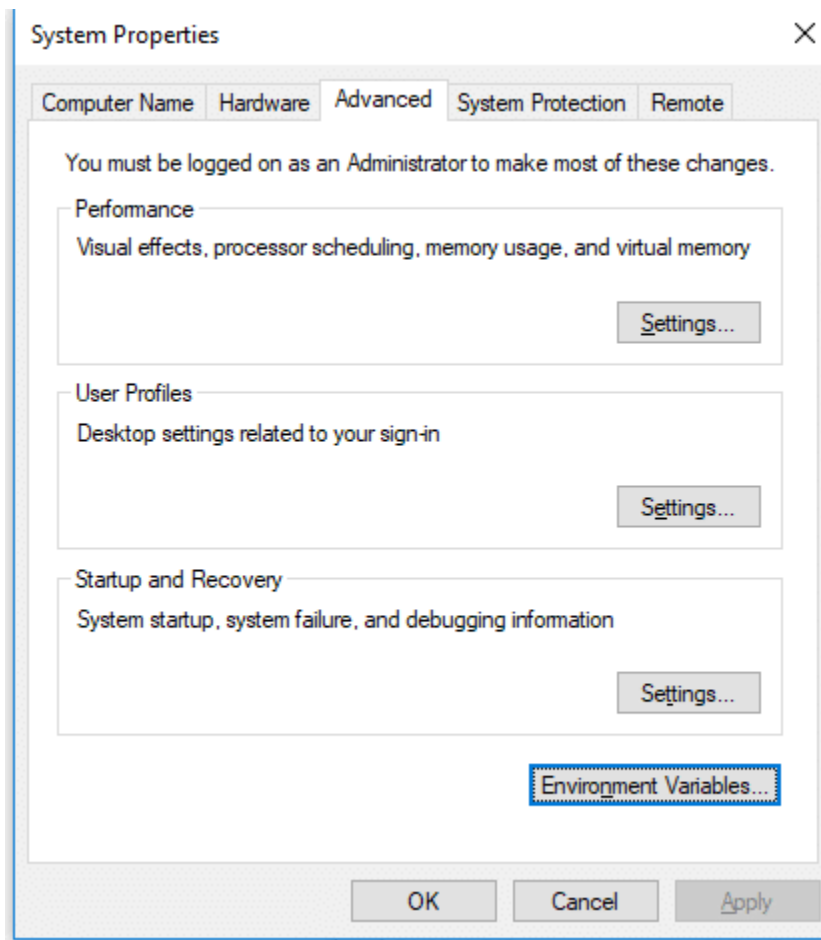
Create a new user variable. Put the Variable_name as HADOOP_HOME and Variable_value as the path of the bin folder where you extracted hadoop.
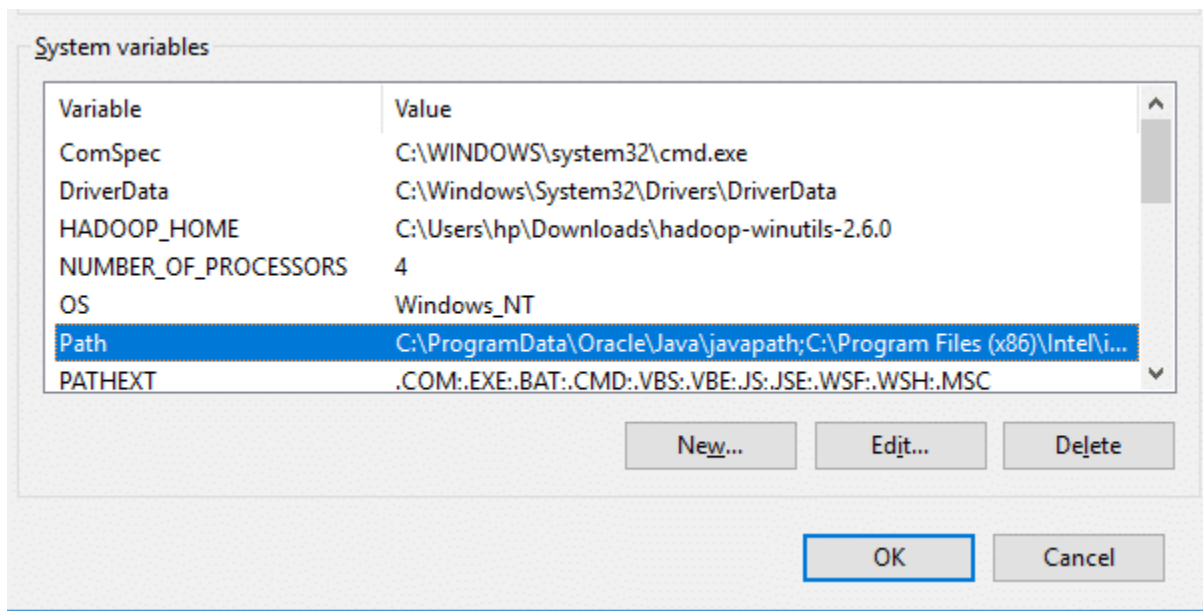

Likewise, create a new user variable with variable name as JAVA_HOME and variable value as the path of the bin folder in the Java directory.
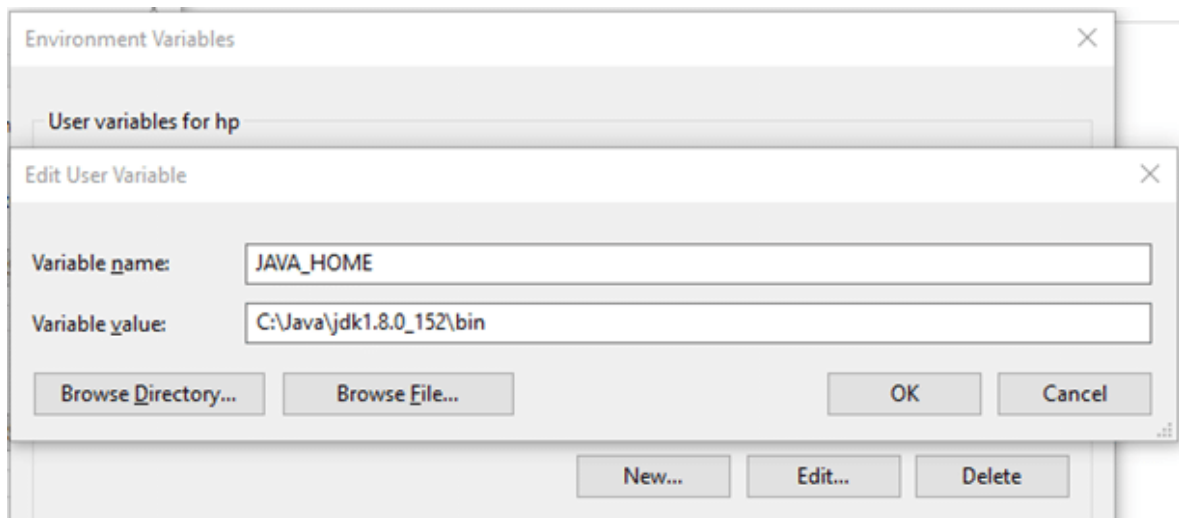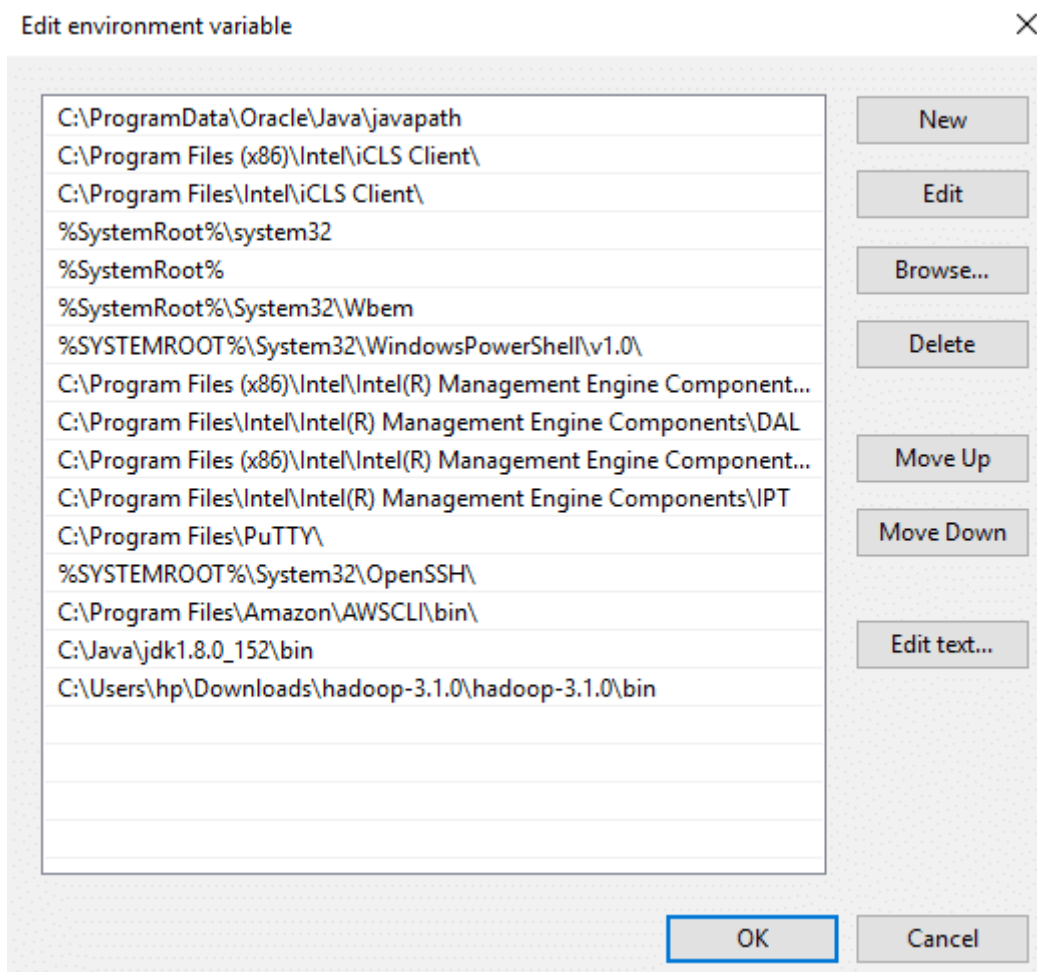
Now we need to set Hadoop bin directory and Java bin directory path in system variable path.

Edit Path in system variable

Click on New and add the bin directory path of Hadoop and Java in it.

Configurations

Now we need to edit some files located in the hadoop directory of the etc folder where we installed hadoop. The files that need to be edited have been highlighted.

Name	Date modified	Type	Size
container-executor.cfg	3/30/2018 5:45 AM	CFG File	2 KB
core-site	3/30/2018 5:31 AM	XML Document	1 KB
hadoop-env	3/30/2018 5:31 AM	Windows Comma...	4 KB
hadoop-env.sh	3/30/2018 5:52 AM	SH File	16 KB
hadoop-metrics2.properties	3/30/2018 5:31 AM	PROPERTIES File	4 KB
hadoop-policy	3/30/2018 5:31 AM	XML Document	11 KB
hadoop-user-functions.sh.example	3/30/2018 5:31 AM	EXAMPLE File	4 KB
hdfs-site	3/30/2018 5:33 AM	XML Document	1 KB
httpfs-env.sh	3/30/2018 5:33 AM	SH File	2 KB
httpfs-log4j.properties	3/30/2018 5:33 AM	PROPERTIES File	2 KB
httpfs-signature.secret	3/30/2018 5:33 AM	SECRET File	1 KB
httpfs-site	3/30/2018 5:33 AM	XML Document	1 KB
kms-acls	3/30/2018 5:31 AM	XML Document	4 KB
kms-env.sh	3/30/2018 5:31 AM	SH File	2 KB
kms-log4j.properties	3/30/2018 5:31 AM	PROPERTIES File	2 KB
kms-site	3/30/2018 5:31 AM	XML Document	1 KB
log4j.properties	3/30/2018 5:31 AM	PROPERTIES File	14 KB
mapred-env	3/30/2018 5:44 AM	Windows Comma...	1 KB
mapred-env.sh	3/30/2018 5:44 AM	SH File	2 KB
mapred-queues.xml.template	3/30/2018 5:44 AM	TEMPLATE File	5 KB
mapred-site	3/30/2018 5:44 AM	XML Document	1 KB
ssl-client.xml.example	3/30/2018 5:31 AM	EXAMPLE File	3 KB
ssl-server.xml.example	3/30/2018 5:31 AM	EXAMPLE File	3 KB
user_ec_policies.xml.template	3/30/2018 5:33 AM	TEMPLATE File	3 KB
workers	3/30/2018 5:31 AM	File	1 KB
yarn-env	3/30/2018 5:43 AM	Windows Comma...	3 KB
yarn-env.sh	3/30/2018 5:43 AM	SH File	6 KB
yarnservice-log4j.properties	3/30/2018 5:43 AM	PROPERTIES File	3 KB
yarn-site	3/30/2018 5:43 AM	XML Document	1 KB

1. Edit the file core-site.xml in the hadoop directory. Copy this xml property in the configuration in the file

```
1 <configuration>
2    <property>
3        <name>fs.defaultFS</name>
4        <value>hdfs://localhost:9000</value>
5    </property>
6 </configuration>
```

2. Edit mapred-site.xml and copy this property in the cofiguration

```
1 <configuration>
2    <property>
3        <name>mapreduce.framework.name</name>
```

```
4      <value>yarn</value>
5    </property>
6 </configuration>
```

3. Create a folder 'data' in the hadoop directory

Create a folder with the name 'datanode' and a folder 'namenode' in this data directory

4. Edit the file hdfs-site.xml and add below property in the configuration

Note: The path of namenode and datanode across value would be the path of the datanode and namenode folders you just created.

```
1  <configuration>
2    <property>
3      <name>dfs.replication</name>
4      <value>1</value>
```

```
5    </property>
6    <property>
7        <name>dfs.namenode.name.dir</name>
8        <value>C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\data\namenode</value>
9    </property>
10   <property>
11       <name>dfs.datanode.data.dir</name>
12       <value> C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\data\datanode</value>
13   </property>
14 </configuration>
```

5. Edit the file yarn-site.xml and add below property in the configuration

```
1   <configuration>
2     <property>
3       <name>yarn.nodemanager.aux-services</name>
4       <value>mapreduce_shuffle</value>
5     </property>
6     <property>
7         <name>yarn.nodemanager.auxservices.mapreduce.shuffle.class</name>
8   <value>org.apache.hadoop.mapred.ShuffleHandler</value>
9     </property>
10 </configuration>
```

6. Edit hadoop-env.cmd and replace %JAVA_HOME% with the path of the java folder
where your jdk 1.8 is installed

```
hadoop-env - Notepad

File  Edit  Format  View  Help

@rem Set Hadoop-specific environment variables here.

@rem The only required environment variable is JAVA_HOME.  All others are
@rem optional.  When running a distributed configuration it is best to
@rem set JAVA_HOME in this file, so that it is correctly defined on
@rem remote nodes.

@rem The java implementation to use.  Required.
set JAVA_HOME=C:\Java\jdk1.8.0_152

@rem The jsvc implementation to use. Jsvc is required to run secure datanodes.
@rem set JSVC_HOME=%JSVC_HOME%

@rem set HADOOP_CONF_DIR=

@rem Extra Java CLASSPATH elements.  Automatically insert capacity-scheduler.
if exist %HADOOP_HOME%\contrib\capacity-scheduler (
  if not defined HADOOP_CLASSPATH (
    set HADOOP_CLASSPATH=%HADOOP_HOME%\contrib\capacity-scheduler\*.jar
  ) else (
    set HADOOP_CLASSPATH=%HADOOP_CLASSPATH%;%HADOOP_HOME%\contrib\capacity-scheduler\*.jar
  )
)
```

Hadoop needs windows OS specific files which does not come with default download of hadoop.

To include those files, replace the bin folder in hadoop directory with the bin folder provided in this github link.

https://github.com/s911415/apache-hadoop-3.1.0-winutils

Download it as zip file. Extract it and copy the bin folder in it. If you want to save the old bin folder, rename it like bin_old and paste the copied bin folder in that directory.

Check whether hadoop is successfully installed by running this command on cmd-

 1 hadoop version

Since it doesn't throw error and successfully shows the hadoop version, that means hadoop is successfully installed in the system.

Format the NameNode

Formatting the NameNode is done once when hadoop is installed and not for running hadoop filesystem, else it will delete all the data inside HDFS. Run this command-

1 hdfs namenode –format

It would appear something like this –

```
C:\Users\hp>hdfs namenode -format
2019-04-07 21:12:06,653 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = DESKTOP-S9AL7B0/
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 3.1.0
```
```
2019-04-07 21:12:08,941 INFO snapshot.SnapshotManager: SkipList is disabled
2019-04-07 21:12:08,941 INFO util.GSet: Computing capacity for map cachedBlocks
2019-04-07 21:12:08,941 INFO util.GSet: VM type       = 64-bit
2019-04-07 21:12:08,941 INFO util.GSet: 0.25% max memory 889 MB = 2.2 MB
2019-04-07 21:12:08,941 INFO util.GSet: capacity      = 2^18 = 262144 entries
2019-04-07 21:12:08,957 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets = 10
2019-04-07 21:12:08,957 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.users = 10
2019-04-07 21:12:08,973 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
2019-04-07 21:12:08,973 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
2019-04-07 21:12:08,973 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry exp
2019-04-07 21:12:08,973 INFO util.GSet: Computing capacity for map NameNodeRetryCache
2019-04-07 21:12:08,973 INFO util.GSet: VM type       = 64-bit
2019-04-07 21:12:08,988 INFO util.GSet: 0.0299999993329447746% max memory 889 MB = 273.1 KB
2019-04-07 21:12:08,988 INFO util.GSet: capacity      = 2^15 = 32768 entries
2019-04-07 21:12:13,586 INFO namenode.FSImage: Allocated new BlockPoolId: BP-1773702794-192.168.56.1-1554651733554
2019-04-07 21:12:13,696 INFO common.Storage: Storage directory C:\hadoop-3.1.0\data\namenode has been successfully fo
2019-04-07 21:12:13,727 INFO namenode.FSImageFormatProtobuf: Saving image file C:\hadoop-3.1.0\data\namenode\current\
on
2019-04-07 21:12:13,887 INFO namenode.FSImageFormatProtobuf: Image file C:\hadoop-3.1.0\data\namenode\current\fsimage
n 0 seconds .
2019-04-07 21:12:14,046 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2019-04-07 21:12:14,062 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at DESKTOP-S9AL7B0/
************************************************************/

C:\Users\hp>
```

Now change the directory in cmd to sbin folder of hadoop directory with this command,

(Note: Make sure you are writing the path as per your system)

1 cd C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin

Start namenode and datanode with this command –

1 start-dfs.cmd

```
C:\Users\hp>cd C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin

C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>start-dfs.cmd

C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>
```

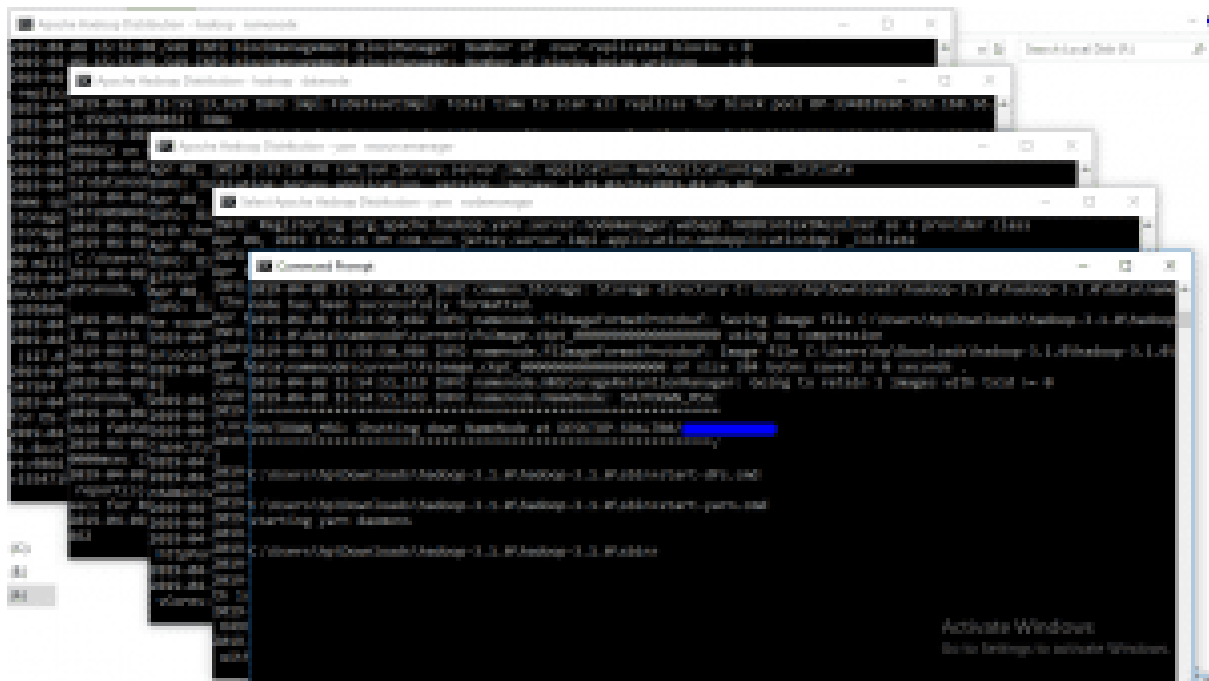Two more cmd windows will open for NameNode and DataNode

Now start yarn through this command-

 1 start-yarn.cmd

```
C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>start-yarn.cmd
starting yarn daemons

C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>
```

Two more windows will open, one for yarn resource manager and one for yarn node manager.

Note: Make sure all the 4 Apache Hadoop Distribution windows are up n running. If they are not running, you will see an error or a shutdown message. In that case, you need to debug the error.

To access information about resource manager current jobs, successful and failed jobs, go to this link in browser-

http://localhost:8088/cluster

To check the details about the hdfs (namenode and datanode),

Open this link on browser-

http://localhost:9870/

Note: If you are using Hadoop version prior to 3.0.0 – Alpha 1, then use port http://localhost:50070/

Working with HDFS

I will be using a small text file in my local file system. To put it in hdfs using hdfs command line tool.

I will create a directory named 'sample' in my hadoop directory using the following command-

 1 hdfs dfs –mkdir /sample

```
C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>hdfs dfs -mkdir /sample

C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>
```

To verify if the directory is created in hdfs, we will use 'ls' command which will list the files present in hdfs –

 1 hdfs dfs –ls /

```
C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>hdfs dfs -ls /
Found 1 items
drwxr-xr-x    - hp supergroup          0 2019-04-07 23:39 /sample

C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>
```

Then I will copy a text file named 'potatoes' from my local file system to this folder that I just created in hdfs using copyFromLocal command-

 1 hdfs dfs -copyFromLocal C:\Users\hp\Downloads\potatoes.txt /sample

```
C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>hdfs dfs -copyFromLocal C:\Users\hp\Downloads\potatoes.txt /sample
C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>
```

To verify if the file is copied to the folder, I will use 'ls' command by specifying the folder name which will read the list of files in that folder –

 1 hdfs dfs –ls /sample

```
C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>hdfs dfs -ls /sample
Found 1 items
-rw-r--r--   1 hp supergroup       3736 2019-04-07 23:39 /sample/potatoes.txt

C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>
```

To view the contents of the file we copied, I will use cat command-

1 hdfs dfs –cat /sample/potatoes.txt

```
C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>hdfs dfs -cat /sample/potatoes.txt
area    temp    size    storage method texture flavor  moistness
1       1       1       1       1       2.9     3.2     3.0
1       1       1       1       2       2.3     2.5     2.6
1       1       1       1       3       2.5     2.8     2.8
1       1       1       1       4       2.1     2.9     2.4
1       1       1       1       5       1.9     2.8     2.2
1       1       1       2       1       1.8     3.0     1.7
1       1       1       2       2       2.6     3.1     2.4
1       1       1       2       3       3.0     3.0     2.9
1       1       1       2       4       2.2     3.2     2.5
1       1       1       2       5       2.0     2.8     1.9
1       1       1       3       1       1.8     2.6     1.5
1       1       1       3       2       2.0     2.8     1.9
1       1       1       3       3       2.6     2.6     2.6
1       1       1       3       4       2.1     3.2     2.1
1       1       1       3       5       2.5     3.0     2.1
1       1       1       4       1       2.6     3.1     2.4
```

To Copy file from hdfs to local directory, I will use get command –

1 hdfs dfs -get /sample/potatoes.txt C:\Users\hp\Desktop\priyanka

```
C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>hdfs dfs -get /sample/potatoes.txt C:\Users\hp\Desktop\priyanka

C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\sbin>
```