HADOOP & SPARK SYSTEM

Hadoop is an open-source framework that allows to store and process big data, in a distributed environment across clusters of computers. Hadoop is designed to scale up from a single server to thousands of machines, where every machine is offering local computation and storage. Spark is an open-source cluster computing designed for fast computation. It provides an interface for programming entire clusters with implicit data parallelism and fault tolerance. The main feature of Spark is in-memory cluster computing that increases the speed of an application.

HADOOP

- **Hadoop is a registered trademark of the Apache software foundation. It utilizes a simple programming model to perform the required operation among clusters. All modules in Hadoop are designed with a fundamental assumption that hardware failures are common occurrences and should be dealt with by the framework.**
- **It runs the application using the MapReduce algorithm, where data is processed in parallel on different CPU nodes. In other words, the Hadoop framework is capable enough to develop applications, which are further capable of running on clusters of computers and they could perform a complete statistical analysis for a huge amount of data.**
- **The core of Hadoop consists of a storage part, which is known as Hadoop Distributed File System and a processing part called the MapReduce**

programming model. Hadoop basically split files into the large blocks and distribute them across the clusters, transfer package code into nodes to process data in parallel.

- This approach dataset to be processed faster and more efficiently. Other Hadoop modules are Hadoop common, which is a bunch of Java libraries and utilities returned by Hadoop modules. These libraries provide a file system and operating system level abstraction, also contain required Java files and scripts to start Hadoop. Hadoop Yarn is also a module, which is being used for job scheduling and cluster resource management.

SPARK

- Spark was built on the top of Hadoop MapReduce module and it extends the MapReduce model to efficiently use more type of computations which include Interactive Queries and Stream Processing. Spark was introduced by the Apache software foundation, to speed up the Hadoop computational computing software process.
- Spark has its own cluster management and is not a modified version of Hadoop. Spark utilizes Hadoop in two ways – one is storage and second is

processing. Since cluster management is arriving from Spark itself, it uses Hadoop for storage purposes only.

- Spark is one of the Hadoop's subprojects which was developed in 2009, and later it became open source under a BSD license. It has lots of wonderful features, by modifying certain modules and incorporating new modules. It helps run an application in a Hadoop cluster, multiple times faster in memory.
- This is made possible by reducing the number of read/write operations to disk. It stores the intermediate processing data in memory, saving read/write operations. Spark also provides built-in APIs in Java, Python or Scala. Thus, one can write applications in multiple ways. Spark not only provides a Map and Reduce strategy but also support SQL queries, Streaming data, Machine learning and Graph Algorithms.

Whereas Hadoop reads and writes files to HDFS, Spark processes data in RAM using a concept known as an RDD, Resilient Distributed Dataset. Spark can run either in stand-alone mode, with a Hadoop cluster serving as the data source, or in conjunction with Mesos. ... Java is another option for writing Spark jobs.

Spark can never be a replacement for Hadoop! Spark is a processing engine that functions on top of the Hadoop ecosystem. Both Hadoop and Spark have their own advantages. Spark is built to increase the processing speed of the Hadoop ecosystem and to overcome the limitations of MapReduce.

INSTALL & RUN HADOOP WITH SPARK ON WINDOWS

SETUP & INSTALATION STEPS

STEP 1: PRE-REQUISITES

Download and install the Following:

- **Download & Install JDK latest version**

  https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

- **Download Hadoop 2.7.2 files**

  https://archive.apache.org/dist/hadoop/core/hadoop-2.7.2/hadoop-2.7.2.tar.gz

- **Download & Install Scala 2.11.8.msi**

  https://downloads.lightbend.com/scala/2.11.8/scala-2.11.8.msi

- **Download Spark 2.3.0 Folder**

  https://archive.apache.org/dist/spark/spark-2.3.0/spark-2.3.0-bin-hadoop2.7.tgz

STEP 2: FOLDER CONFIGURATIONS

Copy all the installation folders to c:\work from the installed paths C:\Program Files

- **C:\work\hadoop-2.7.2**
- **C:\work\scala**
- **C:\work\java\jdk1.8.0_191**

- C:\work\spark-2.3.0-bin-hadoop2.7

Create Empty Folders:

- C:\tmp\hive
- C:\work\hadoop272data\datanode
- C:\work\hadoop272data\namenode

STEP 3: SETTING ENVIRONMENT VARIABLES

- Set Environment variables (Replace or Remove Old/Earlier Configurations)User Variables:
  JAVA_HOME C:\work\java\jdk1.8.0_191 (based on the version of JDK downloaded)
  HADOOP_HOME C:\work\hadoop-2.7.2
  SPARK_HOME C:\work\spark-2.3.0-bin-hadoop2.7
  SCALA_HOME C:\work\scala
- System Variables for Path:
  Path
  - C:\work\scala\bin
    C:\work\spark-2.3.0-bin-hadoop2.7\bin
    C:\work\hadoop-2.7.2\bin
    C:\work\hadoop-2.7.2\sbin
- Download Windows 7/8/10 pre-configured files from [http://www.praveenkumarg.com/wp content/uploads/2018/12/hadoop2.7.2.zip](http://www.praveenkumarg.com/wp-content/uploads/2018/12/hadoop2.7.2.zip)
  – Delete Folders C:\work\hadoop-2.7.2\bin and C:\work\hadoop-2.7.2\etc
  – Replace bin and etc. folders from downloaded .zip file.

edit the file and SET JAVA_HOME path in C:\work\hadoop-2.7.2\etc\hadoop\hadoop-env.sh

set JAVA_HOME=C:\work\Java\jdk1.8.0_191 (based on the version of JDK downloaded)

STEP 4: VALIDATE CONFIGURATIONS FOR HADOOP NODES

Set the following Configuration at C:\work\hadoop-2.7.2\etc\hadoop\hdfs-site.xml

```
<configuration>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value>/C:/work/hadoop272data/namenode</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
    <value>/C:/work/hadoop272data/datanode</value>
</property>
</configuration>
```

STEP 5 : COMMANDA TO RUN HADOOP & SPARK

Execute following commands (cmd.exe) Step by step

1. **> hdfs.cmd namenode -format**
2. **> start-dfs.cmd && start-yarn.cmd (Note: if any popup appears, press allow to access)**
3. **> jps (validate if hadoop is running)**
4. **> cd c:\work\hadoop-2.7.2\bin**
5. **> winutils.exe chmod 777 c:\tmp\hive**
6. **> spark-shell.cmd (to run spark command)**