*This document was written on approx. 20.03.2018*

# Table of contents

[Table of contents 1](#_Toc15819404)

[Prerequisites 2](#_Toc15819405)

[JDK 2](#_Toc15819406)

[Maven 2](#_Toc15819407)

[Cygwin 2](#_Toc15819408)

[Hadoop configuration 14](#_Toc15819409)

# Prerequisites

Validate that JDK, Maven, Cygwin are installed. If not, you should install them (find them in this document).

## JDK

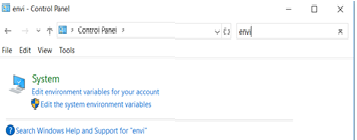
* For instruction on how to install JDK see <https://github.com/alex-ber/AlexBerDocs/blob/master/Working%20Station/Windows/Java%20Env.docx> section JDK

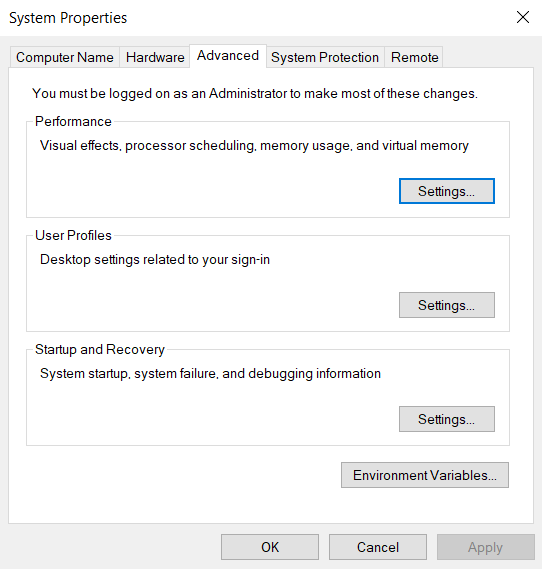
## Maven

* For instruction on how to install Maven see <https://github.com/alex-ber/AlexBerDocs/blob/master/Working%20Station/Windows/Java%20Env.docx> section Maven

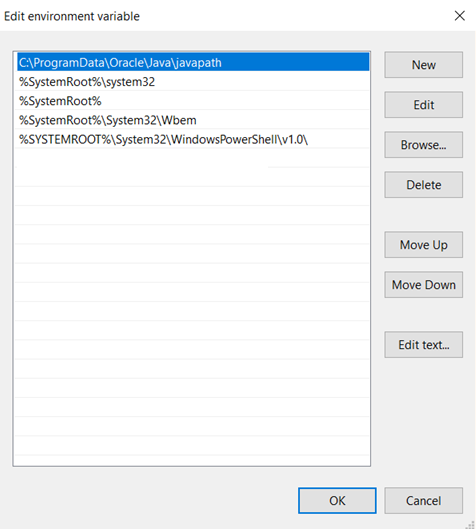
## Cygwin

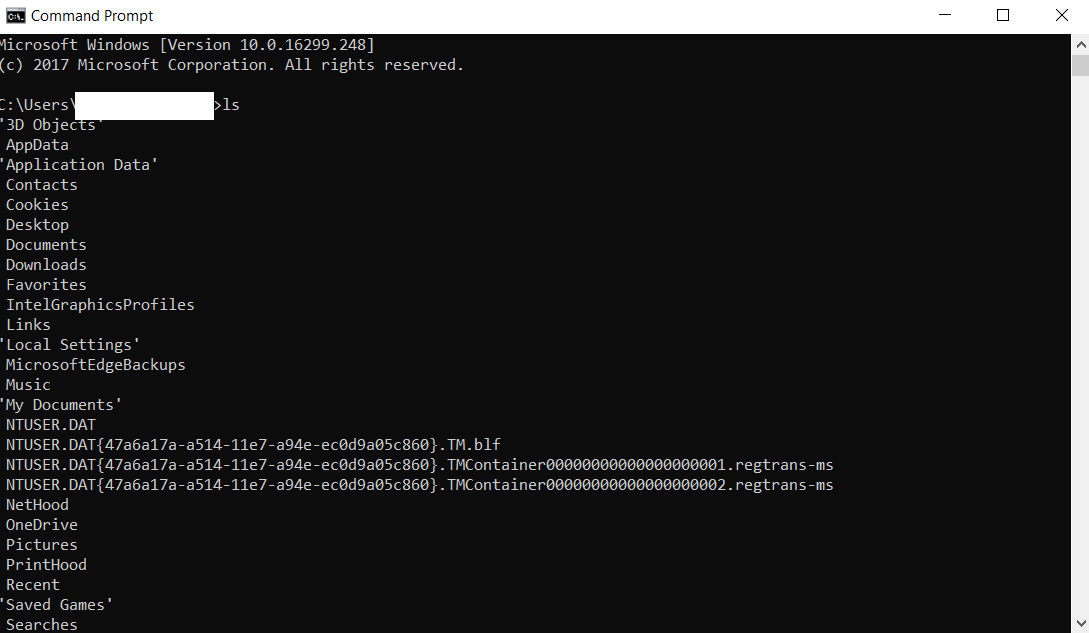
* From <https://www.cygwin.com/install.html> download and install cygwin-setup-x86\_64.exe (This file is for 64-bit version of Windows)
* Go to environment variables

****

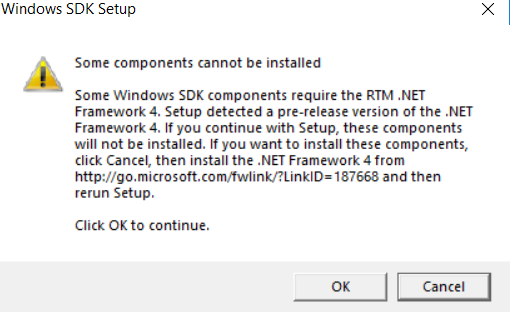


* Click on Environment Variables
* Go to System variables section.
* Search for **Path** variable in the “**System Variable**” section in “Environment Variables” dialogue box you just opened. Click on Edit button.



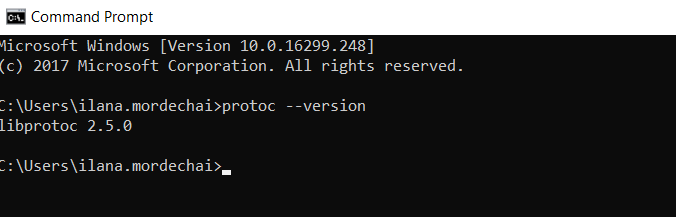
* To **Path** variable in the “**System Variable**” section in “Environment Variables”  
  add c:\cygwin64\bin\
* (Optionally) you can find shortcut Cygwin64 Terminal at C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Cygwin
* Open new cmd terminal and right ls.   
  

~~7.1. We are also going to install .Net Framework 4 in this tutorial. If you have any Visual Studio versions installed on your machine then this is likely to cause issues in the build process because of version mismatch of some .Net components and in some cases will not allow you to install Windows SDK 7.1.~~

* Go to Control Panel\Programs\Programs and Features uninstall all Microsoft Visual C++ Redistributables, if they got installed with the OS because they may be a newer version than the one which Win SDK 7.1 requires. During SDK installation they will cause errors.
* Run winsdk\_web.exe  
  You will see following message  
    
  To **Path** variable in the “**System Variable**” section in “Environment Variables”  
  add c:\cygwin64\bin\ (see above).
*   
  Ignore it (we will install these component manually later). Click ok.
* Install **VC-Compiler-KB2519277.exe (these are missing components).**
* **Put header ammintrin.h** file which needs to be placed in "C:\Program Files (x86)\Microsoft Visual Studio 10.0\VC\INCLUDE\". (See <https://www.mathworks.com/matlabcentral/answers/90383-fix-problem-when-mex-cpp-file?s_tid=gn_loc_drop>)

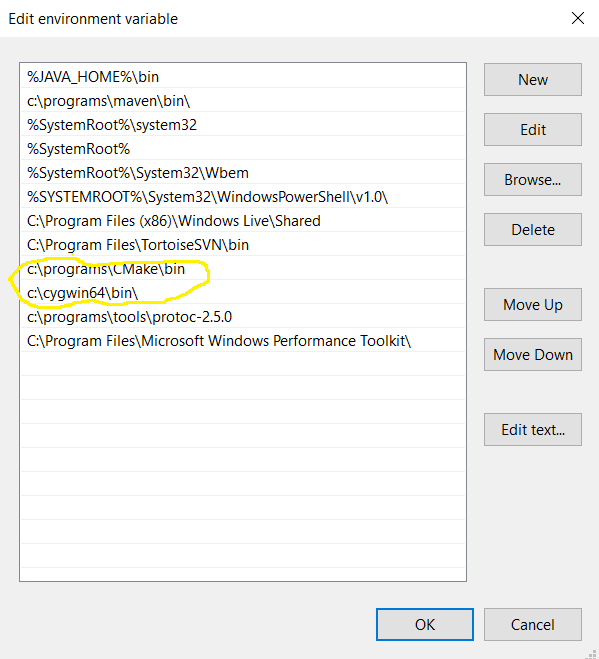
1. ProtocolBuffer 2.5.0

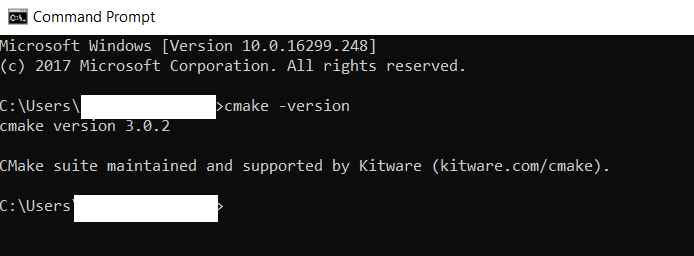
* Extract protoc-2.5.0-win32.zip to c:\programs\tools\protoc-2.5.0
* To **Path** variable in the “**System Variable**” section in “Environment Variables”  
  add c:\programs\tools\protoc-2.5.0 (see above).
* To check if the protocol buffer installation is working fine just type command protoc –version



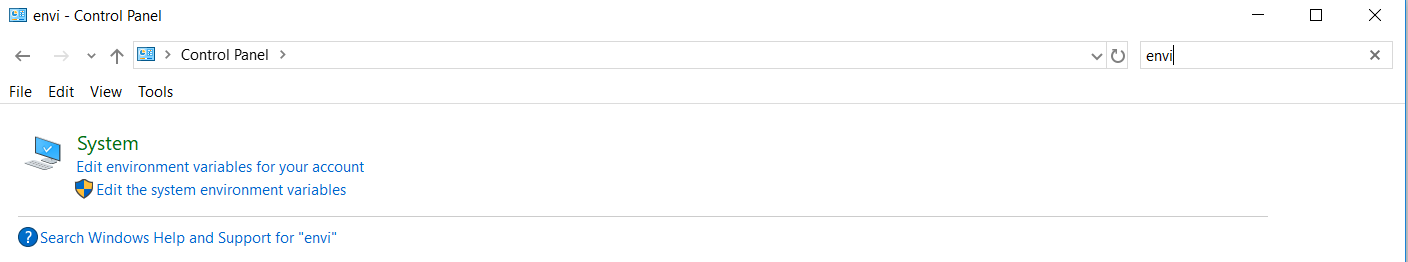
1. CMake 3.5.0

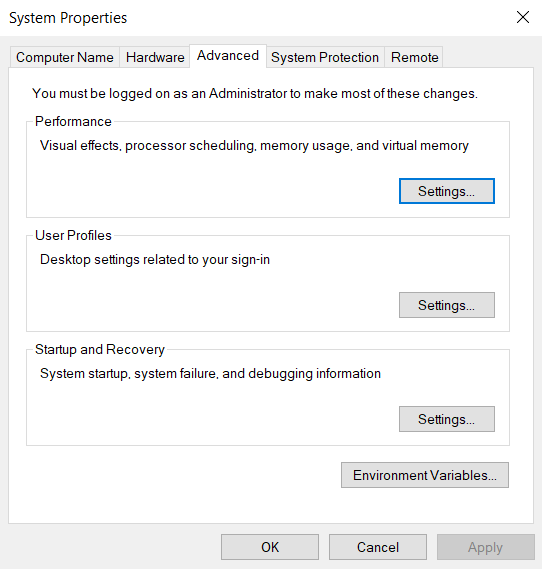
* Install cmake-3.0.2-win32-x86.exe to c:\programs\CMake  
    
  To **Path** variable in the “**System Variable**” section in “Environment Variables”  
  add c:\programs\CMake\bin  
    
  Make sure to use this CMake and not the CMake that comes with cygwin.



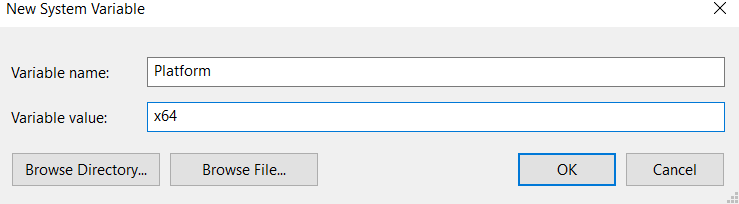
To check that the CMake installation is correct, open a new command prompt and type cmake -version  
  


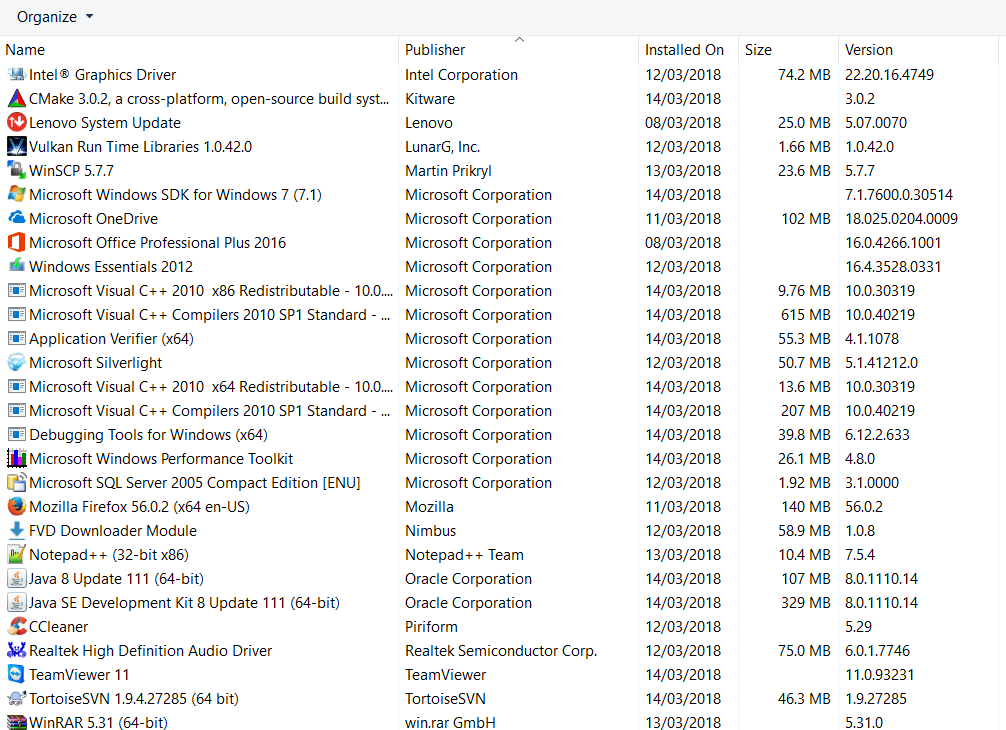
* Go to environment variables

****

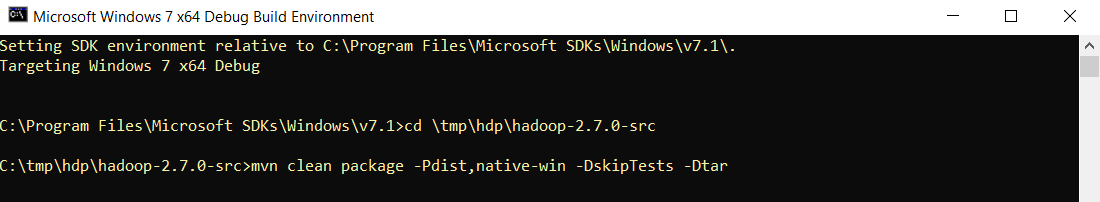


* Click on Environment Variables
* Go to System variables section.
* Create new variable Platform with value **x64**

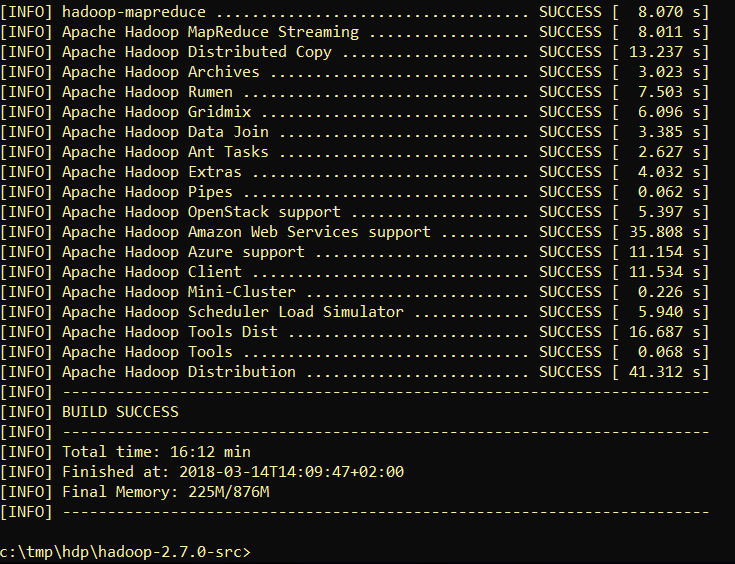


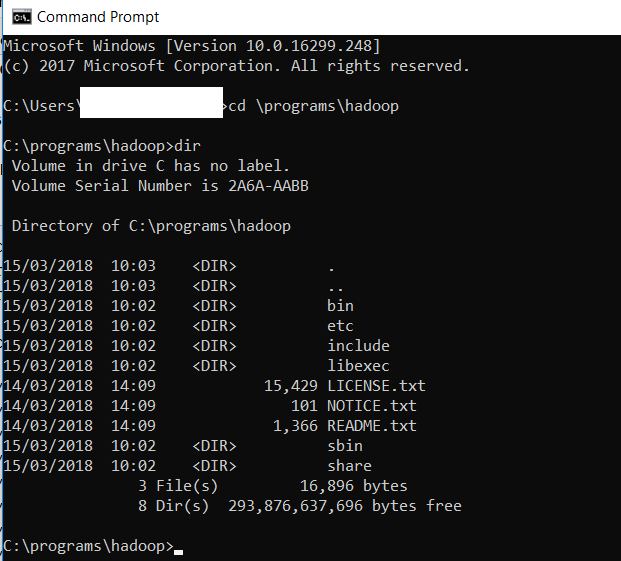
* Click ok
* So before we proceed on the build, just have a look on the state of all installed programs on my machine:  
    
  ****

1. Extract hadoop-2.7.0-src.tar.gz inner archive to some tmp directory (c:\tmp\hdp for example). <http://archive.apache.org/dist/hadoop/common/hadoop-2.7.0/hadoop-2.7.0-src.tar.gz>
2. Open **Windows SDK 7.1 Command Prompt** (Start --> All Programs --> Microsoft Windows SDK v7.1, and click on **Windows SDK 7.1 Command Prompt**).
3. Change the directory to your extracted Hadoop source folder. For this tutorial its C:\hdp by typing command ***cd c:\tmp\hdp\hadoop-2.7.0-src***
4. **Validate that you are connected to Israel network and you take settings.xml from Box.**
5. Now type command ***mvn clean package -Pdist,native-win -DskipTests -Dtar***



1. If everything goes well you will see a success message like the below image. Your native Hadoop distribution will be created at c:\tmp\hdp\hadoop-2.7.0-src\hadoop-dist\target\

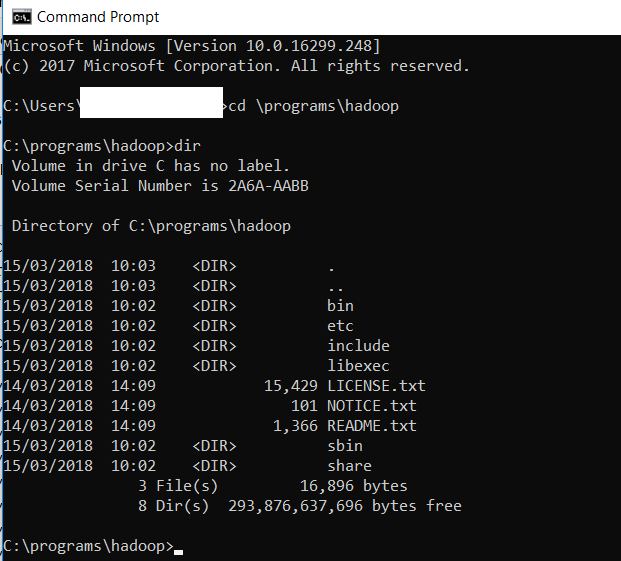


1. In c:\tmp\hdp\hadoop-2.7.0-src\hadoop-dist\target\hadoop-2.7.0 you can find “hadoop-2.7.0.tar.gz”. This Hadoop distribution contains native Windows binaries and can be used on a Windows OS for Hadoop clusters.
2. Extract tar.gz to c:\programs\hadoop. This directory should look like  
     
   
3. For running a Hadoop instance you need to change some configuration files like **hadoop-env.cmd, core-site.xml,hdfs-site.xml, slaves, etc.** For those changes please follow this official link to setup and run hadoop on windows: [https://wiki.apache.org/hadoop/Hadoop2OnWindows.](https://wiki.apache.org/hadoop/Hadoop2OnWindows)

# Hadoop configuration

Inspired by <https://wiki.apache.org/hadoop/Hadoop2OnWindows>

Note: The link was changed <https://cwiki.apache.org/confluence/display/HADOOP2/Hadoop2OnWindows>

* This document describes configuration and installing version 2.7.0 – Download hadoop-2.7.0-src.tar.gz from <https://archive.apache.org/dist/hadoop/core/hadoop-2.7.0/>
* I assume that you have extracted tar.gz to to c:\programs\hadoop. This directory should look like  
    
    
    
  
* Now we will configure Hadoop as Single Node (pseudo-distributed) Cluster.  
    
  The configuration file templates will all be found in c:\programs\hadoop\etc\hadoop
* Edit the file **hadoop-env.cmd** to add the following lines near the end of the file.

First edit the file **hadoop-env.cmd** to add the following lines near the end of the file.

set HADOOP\_PREFIX=c:\programs\hadoop

set HADOOP\_CONF\_DIR=%HADOOP\_PREFIX%\etc\hadoop

set YARN\_CONF\_DIR=%HADOOP\_CONF\_DIR%

set PATH=%PATH%;%HADOOP\_PREFIX%\bin

* Edit or create the file **core-site.xml** and make sure it has the following configuration key:

<configuration>

<property>

<name>fs.default.name</name>

<value>hdfs://0.0.0.0:19000</value>

</property>

</configuration>

* Edit or create the file **hdfs-site.xml** and add the following configuration key:

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

</configuration>

* Finally, edit or create the file **slaves** and make sure it has the following entry:

localhost

* Edit or create **mapred-site.xml** under **%HADOOP\_PREFIX%\etc\hadoop** and add the following entries, replacing %USERNAME% with your Windows user name.

<configuration>

<property>

<name>mapreduce.job.user.name</name>

<value>%USERNAME%</value>

</property>

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

<property>

<name>yarn.apps.stagingDir</name>

<value>/user/%USERNAME%/staging</value>

</property>

<property>

<name>mapreduce.jobtracker.address</name>

<value>local</value>

</property>

</configuration>

* Finally, edit or create **yarn-site.xml** and add the following entries:

<configuration>

<!-- Site specific YARN configuration properties -->

<property>

<name>yarn.server.resourcemanager.address</name>

<value>0.0.0.0:8020</value>

</property>

<property>

<name>yarn.server.resourcemanager.application.expiry.interval</name>

<value>60000</value>

</property>

<property>

<name>yarn.server.nodemanager.address</name>

<value>0.0.0.0:45454</value>

</property>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>

<value>org.apache.hadoop.mapred.ShuffleHandler</value>

</property>

<property>

<name>yarn.server.nodemanager.remote-app-log-dir</name>

<value>/app-logs</value>

</property>

<property>

<name>yarn.nodemanager.log-dirs</name>

<value>/dep/logs/userlogs</value>

</property>

<property>

<name>yarn.server.mapreduce-appmanager.attempt-listener.bindAddress</name>

<value>0.0.0.0</value>

</property>

<property>

<name>yarn.server.mapreduce-appmanager.client-service.bindAddress</name>

<value>0.0.0.0</value>

</property>

<property>

<name>yarn.log-aggregation-enable</name>

<value>true</value>

</property>

<property>

<name>yarn.log-aggregation.retain-seconds</name>

<value>-1</value>

</property>

<property>

<name>yarn.application.classpath</name>

<value>%HADOOP\_CONF\_DIR%,%HADOOP\_COMMON\_HOME%/share/hadoop/common/\*,%HADOOP\_COMMON\_HOME%/share/hadoop/common/lib/\*,%HADOOP\_HDFS\_HOME%/share/hadoop/hdfs/\*,%HADOOP\_HDFS\_HOME%/share/hadoop/hdfs/lib/\*,%HADOOP\_MAPRED\_HOME%/share/hadoop/mapreduce/\*,%HADOOP\_MAPRED\_HOME%/share/hadoop/mapreduce/lib/\*,%HADOOP\_YARN\_HOME%/share/hadoop/yarn/\*,%HADOOP\_YARN\_HOME%/share/hadoop/yarn/lib/\*</value>

</property>

</configuration>

* Initialize Environment Variables

Run  **\hadoop-env.cmd** to setup environment variables that will be used by the startup scripts and the daemons.

* Format the FileSystem

Format the filesystem with the following command:

%HADOOP\_PREFIX%\bin\hdfs namenode –format

* Start HDFS daemon

Run the following command to start the NameNode and DataNode on localhost.

%HADOOP\_PREFIX%\sbin\start-dfs.cmd

* To verify that the HDFS daemons are running, try copying a file to HDFS.

c:\programs\hadoop\etc\hadoop>%HADOOP\_PREFIX%\bin\hdfs dfs -put log4j.properties /

c:\programs\hadoop\etc\hadoop>%HADOOP\_PREFIX%\bin\hdfs dfs -ls /

Found 1 items

-rw-r--r-- 1 %USERNAME supergroup 11237 2018-03-18 09:55 /log4j.properties

* Start YARN Daemon and run a YARN job

Finally, start the YARN daemons.

%HADOOP\_PREFIX%\sbin\start-yarn.cmd

The cluster should be up and running! To verify, we can run a simple wordcount job on the text file we just copied to HDFS.

%HADOOP\_PREFIX%\bin\yarn jar %HADOOP\_PREFIX%\share\hadoop\mapreduce\hadoop-mapreduce-examples-2.7.0.jar wordcount /log4j.properties /out