**Documento de arranque**

**Replicación de la prueba de concepto**

**Creado**: 13 Febrero

**Última modificación**: 5 Abril

Habiendo instalado el sistema operativo Ubuntu en el ordenador personal, se han seguido los siguientes pasos para replicar la prueba de concepto que se diseñó durante el mes de Diciembre de 2017:

**Paso 1.** Se ha descargado el fichero de los productos fitosanitarios autorizados y se ha pasado a formato csv.

**¿Cómo?**

**1.1:** Acceder a la página de los productos fitosanitarios de mapama [[1](http://www.mapama.gob.es/es/agricultura/temas/sanidad-vegetal/productos-fitosanitarios/registro/menu.asp)].

**1.2:** Desde el apartado “Productos fitosanitarios”, descargar el “Listado de productos autorizados“ [[2](http://www.mapama.gob.es/es/agricultura/temas/sanidad-vegetal/productos-fitosanitarios/registro/productos/ListadoProductos.asp)] y guardarlo en formato csv.

**Paso 2.** Se ha instalado Java JDK en el equipo.

**¿Cómo?**

**2.1:** En primer lugar comprobar si Java está instalado en el equipo. Para ello, ejecutar el comando *java -version.* Si no aparece la versión, significa que no está instalado.

**2.2:** Ejecutar el comando *sudo apt-get update* para actualizar el sistema.

**2.3:** Ejecutar el comando *sudo apt-get install openjdk-8-jre* para descargar la versión 8 de java.

**2.4:** Ejecutar otra vez *java -version* para comprobar que efectivamente se ha descargado e instalado java 8.

**Paso 3.** Se ha instalado Spring XD en el equipo.

**¿Cómo?**

**3.1:** Siguiendo la Spring reference guide [[3](http://docs.spring.io/spring-xd/docs/current/reference/html/)] primero se descarga el fichero zip con la última release estable (en mi caso 1.3.1 [[4](http://repo.spring.io/libs-release/org/springframework/xd/spring-xd/1.3.1.RELEASE/spring-xd-1.3.1.RELEASE-dist.zip)]).

**3.2:** Se descomprime dicho fichero.

**3.3:** Para arrancar spring, hay que ejecutar el archivo que se encuentra en: <*ruta\_del fichero spring-xd-1.3.1.RELEASE/xd/bin/xd-singlenode>*

**3.4:** Para arrancar una consola y realizar operaciones con Spring, hay que ejecutar la shell de Spring, que es el archivo ubicado en: <*ruta\_del fichero spring-xd-1.3.1.RELEASE/shell/bin/xd-shell>.*

**Paso 4.** Se ha instalado Hadoop en el equipo.

**¿Cómo?**

**4.1:** Siguiendo el tutorial de Digital Ocean [[5](https://www.digitalocean.com/community/tutorials/how-to-install-hadoop-in-stand-alone-mode-on-ubuntu-16-04)], lo primero que se debe comprobar es que Java esté instalado. Yo lo instalé en el paso 2.

**4.2:** Acceder a la página oficial de Apache Hadoop Releases [[6](http://hadoop.apache.org/releases.html)] y hacer click sobre los binarios de la última versión estable (en mi caso el 2.7.3). Eso nos llevará a otra página donde tendremos que copiar el enlace recomendado (primer enlace disponible).

**4.3:** En una terminal de Ubuntu, ejecutar el comando wget con el link anterior copiado: *wget http://apache.mirrors.tds.net/hadoop/common/hadoop-2.7.3/hadoop-2.7.3.tar.gz*

**4.4:** En este punto, Hadoop está descargado; no obstante, para estar seguros, se debería comprobar que los archivos no están corruptos con el checksum, de la siguiente manera:

**4.4.1**: Acceder a la página de las revisiones oficiales de Hadoop [[7](https://dist.apache.org/repos/dist/release/hadoop/common/)], pinchar en la versión descargada (en mi caso la 2.7.3) y copiar el link del fichero llamado “hadoop-2.7.3.tar.gz.mds” [[8](https://dist.apache.org/repos/dist/release/hadoop/common/hadoop-2.7.3/hadoop-2.7.3.tar.gz.mds)].

**4.4.2**: Descargar dicho fichero con la herramienta wget: *wget https://dist.apache.org/repos/dist/release/hadoop/common/hadoop-2.7.3/hadoop-2.7.3.tar.gz.mds*

**4.4.3**: Con el siguiente comando se realiza la verificación: *shasum -a 256 hadoop-2.7.3.tar.gz*

**4.4.4**: Lo anterior sacará como output una ristra de caracteres junto a la versión de Hadoop descargada.

**4.4.5**: Ahora lo único que queda es comprobar de que el output del paso anterior coincide con la clave que descargada en el punto 4.4.2, que podremos ver con un simple cat: *cat hadoop-2.7.3.tar.gz.mds*. En caso afirmativo, es seguro afirmar que los archivos se han descargado correctamente y no hay ninguno corrupto.

**4.5:** Lo siguiente es descomprimir el archivo descargado (comando: *tar -xzvf hadoop-2.7.3.tar.gz)* y copiarlo descomprimido en la carpeta /usr/local, que es el sitio adecuado para el software local. Esto último se hará con el comando *sudo mv hadoop-2.7.3 /usr/local/hadoop.* En este punto, Hadoop está instalado en el equipo.

**Paso 5.** Se ha configurado la ruta del Java Home para Hadoop.

**¿Cómo?**

**5.1:** En el fichero de entorno de Hadoop, modificar la línea que indica el path de Java por el siguiente comando, que descubre la ruta de Java de forma dinámica: *readlink -f /usr/bin/java | sed "s:bin/java::*

Para ello abrir el fichero de configuración de Hadoop (*sudo nano /usr/local/hadoop/etc/hadoop/hadoop-env.sh)* y comentar la línea *export JAVA\_HOME=${JAVA\_HOME}.* A continuación, debajo, poner lo siguiente: *export JAVA\_HOME=$(readlink -f /usr/bin/java | sed "s:bin/java::").*

**5.2:** En este punto Hadoop debería estar configurado para arrancarlo en cualquier momento. Para probar que efectivamente esto es así, ejecutar */usr/local/hadoop/bin/hadoop,* que indicará la forma correcta de usar Hadoop.

**Paso 6.** Se ha instalado Spring Flo para facilitar la creación de Streams con Spring.

**¿Cómo?**

**6.1:** Acceder a la página de releases de Flo [[9](https://network.pivotal.io/products/p-spring-flo)] y descargar el jar de la GUI correspondiente a Flo (flo-spring-xd-admin-ui-client-1.3.1.RELEASE.zip).

**6.2:** Descargar y descomprimir el zip anterior (jar deseado).

**6.3:** Copiar el jar anterior en la carpeta de *<instalación de Spring/xd/lib>,* sustituyéndolo por el ya existente.

**6.4:** Arrancar Spring y una shell y acceder a la dirección *http://localhost:9393/admin-ui/#/streams/create* Si no se permite esto, borrar la caché del navegador y volver a ejecutar este mismo paso (6.4).

**Paso 7.** Tras comprobar que la versión local de Hadoop funciona, se ha querido ampliarlo para dotarlo de funcionalidades pseudo-distribuidas. Esto es, aunque realmente el trabajo se lleve a cabo en local (localhost), estará preparado para ser lanzado en distribuido. [[10](https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/SingleCluster.html#Pseudo-Distributed_Operation)]

**¿Cómo?**

**7.0:** Opcional: Si ssh no está instalado en el equipo, instalarlo: *sudo apt-get install ssh*

**7.1:** El siguiente paso es configurar Hadoop para que funcione de manera pseudo-distribuida. Para ello hay que modificar los ficheros de configuración del core de hadoop y del sistema de ficheros (hdfs):

**7.1.1:** Ir a la ruta de instalación de Hadoop, y en el fichero etc/hadoop/core-site.xml añadir lo siguiente en el tag *<configuration>:*

*<property>  
 <name>fs.defaultFS</name>  
 <value>hdfs://localhost:9000</value>  
 </property>*

Esto nos permite decirle a Hadoop que su sistema de ficheros estará accesible en localhost:9000.

**7.1.2:** En el fichero etc/hadoop/hdfs-site.xml añadir lo siguiente en el tag <configuration>:

<property>  
 <name>dfs.replication</name>  
 <value>1</value>  
 </property>

Esto permite indicarle a hadoop que el número de réplicas distribuidas que contendrá el sistema de ficheros es 1 (el local, localhost).

**7.2:** En este paso hay que comprobar si ssh localhost permite conectar directamente:

Ejecutar *ssh localhost.* En caso de pedir contraseña, hay que ejecutar los siguientes pasos:

**7.2.1**: *ssh-keygen -t rsa* (Pulsar enter en cada línea de output que saque este comando)

**7.2.2**: *cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys*

**7.2.3**:*chmod og-wx ~/.ssh/authorized\_keys*

**7.2.4**: Comprobar con *ssh localhost* que ahora no se requiere una contraseña al intentar conectar a localhost con ssh. La salida debería ser algo del estilo: Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-64-generic x86\_64) …

**7.3:** A continuación se intentará ejecutar un job de MapReduce. Para ello, lo primero que hay que hacer es formatear el sistema de ficheros de Hadoop: *hdfs namenode -format*

**7.4:** Ahora se debe arrancar el nodo creado con el sistema de ficheros anteriormente formateado. Para ello, se lanza la siguiente herramienta, ubicada en la carpeta sbin:

*sbin/start-dfs.sh*. Para comprobar que realmente el servicio está bien lanzado, además de comprobar que no saltan errores en la consola, se puede acceder a <http://localhost:50070> y ver que realmente hay un nodo (Datanodes) arrancado. Otra manera es ejecutando el comando “jps” que lista los daemons que están corriendo. Si aparece un demonio llamado “Datanode” y otro llamado “Dataname”, es que el proceso se ha realizado correctamente.

**7.5:** Se deben crear los directorios de HDFS necesarios para ejecutar un job de MapReduce. Para ello se crean los directorios user y user/<username>:

*hdfs dfs -mkdir /user*

*hdfs dfs -mkdir /user/<username>*

**7.6:** Para realizar la prueba, copiar los ficheros de etc/hadoop (local) en HDFS (remoto -localhost) como nombre “input”: *hdfs dfs -put etc/hadoop input*

**7.7:** Ahora efectívamente se puede probar la herramienta MapReduce con la carpeta input creada en el punto anterior:

*hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.3.jar grep input output 'dfs[a-z.]+'*

**7.8:** Para ver los resultados, ejecutar el siguiente comando: *hdfs dfs -cat output/\** ó se puede traer la carpeta output a local y verla desde local:

*hdfs dfs -get output output*

*cat output/\**

**7.9:** Para parar el proceso lanzado (o el nodo en ejecución): *sbin/stop-dfs.sh*

[1]: [Fitosanitarios mapama](http://www.mapama.gob.es/es/agricultura/temas/sanidad-vegetal/productos-fitosanitarios/registro/menu.asp)

[2]: [Listado de Productos Autorizados (Excel)](http://www.mapama.gob.es/es/agricultura/temas/sanidad-vegetal/productos-fitosanitarios/registro/productos/ListadoProductos.asp)

[3]: [Spring reference guide](http://docs.spring.io/spring-xd/docs/current/reference/html/)

[4]: [Spring XD release 1.3.1](http://repo.spring.io/libs-release/org/springframework/xd/spring-xd/1.3.1.RELEASE/spring-xd-1.3.1.RELEASE-dist.zip)

[5]: [Hadoop installation on Ubuntu 16.04](https://www.digitalocean.com/community/tutorials/how-to-install-hadoop-in-stand-alone-mode-on-ubuntu-16-04)

[6]: [Apache Hadoop releases](http://hadoop.apache.org/releases.html)

[7]: [Apache Hadoop revisions](https://dist.apache.org/repos/dist/release/hadoop/common/hadoop-2.7.3)

[8]: [Hadoop 2.7.3 checksum](https://dist.apache.org/repos/dist/release/hadoop/common/hadoop-2.7.3/hadoop-2.7.3.tar.gz.mds)

[9]: [Spring Flo Release](https://network.pivotal.io/products/p-spring-flo)

[10]: [Pseudo-distributed Hadoop setup](https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/SingleCluster.html#Pseudo-Distributed_Operation)