Universidad Técnica Particular de Loja

Arquitectura y Computación Paralela

Por:

√ Vanessa Sotomayor

√ Jairo Valle Roman

√ John Villavicencio

Tutor:

Ing. Gladys Tenesaca

Titulación:

Sistemas Informáticos y Computación

Tema:

Anexo: Manual de Instalación Apache Spark

Período Académico

Oct 2016 – Feb2016

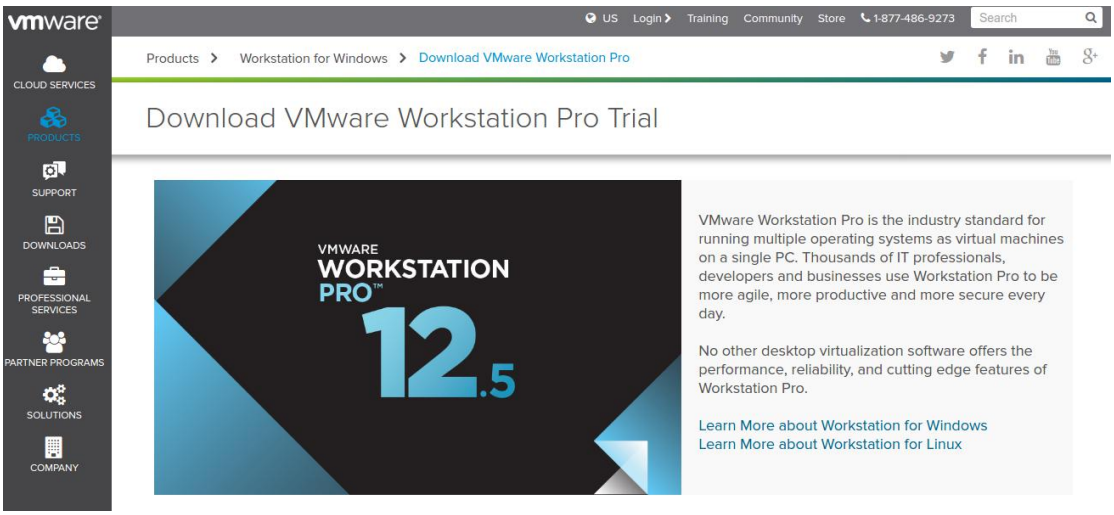
Manual de Instalación Apache Spark 1.4.0

El presente manual tiene como objetivo indicar paso a paso como se instala y configura la herramienta Apache Spark para crear un cluster de computadores. En este manual se utilizara maquinas virtuales para la demostración.

1. Creación de maquina virtual

a) Descarga e instalación de VMware workstation Pro

La descarga se la puede realizar de manera gratuita desde su pagina oficial:
http://www.vmware.com/products/workstation/workstation-evaluation.html

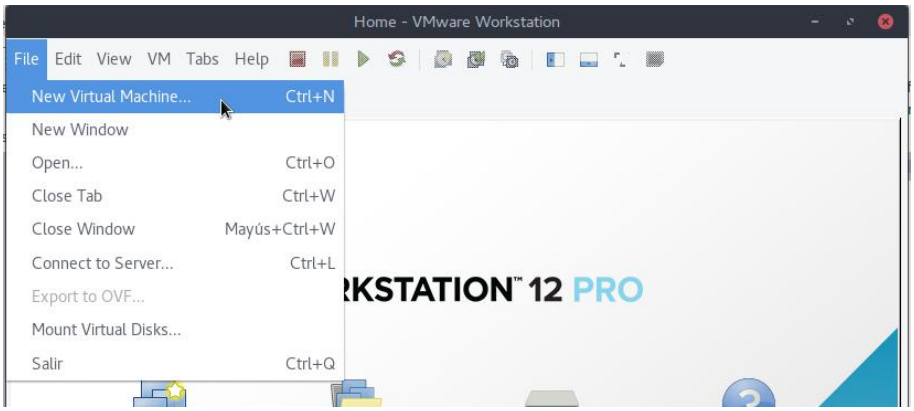


Una vez descargado el programa basta con ejecutarlo y seguir el asistente de instalación que incorpora.

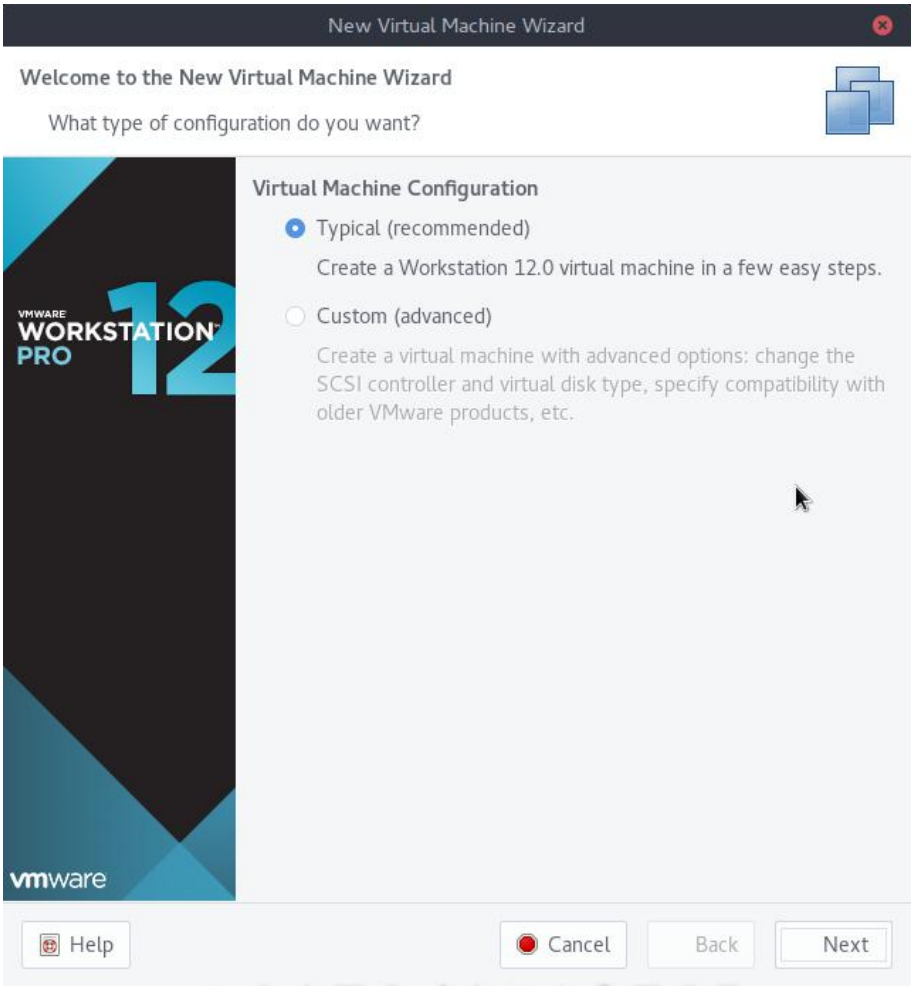
b) Crear primera maquina virtual con Ubuntu 16.04

Nota: En un inicio crearemos una sola maquina virtual en la que instalaremos todos los requisitos de software necesario, posteriormente clonaremos esta maquina virtual para la interconexión.

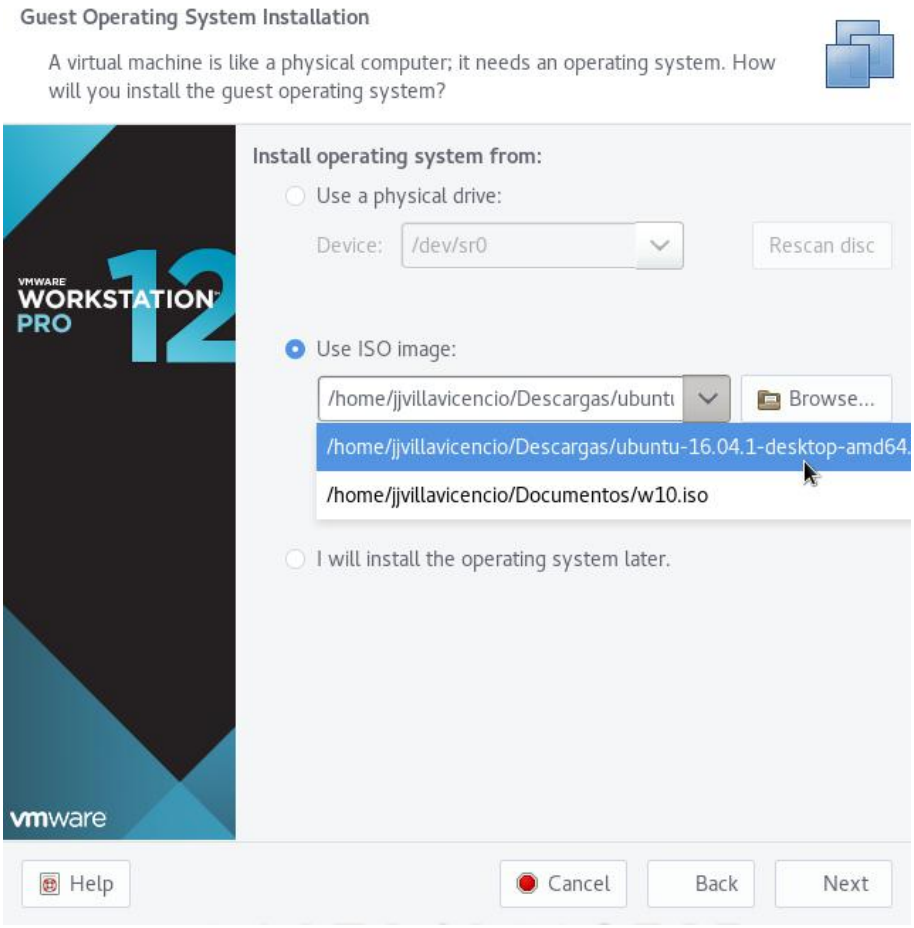
Dentro de la aplicación vamos a **File>New Virtual Machine**



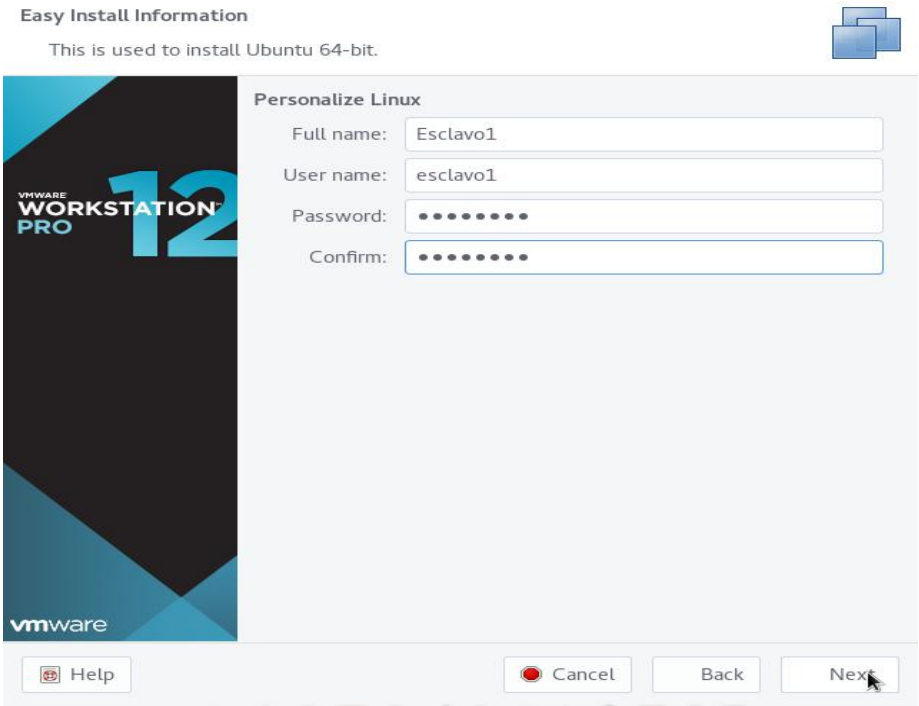
En el asistente escogemos **Typical** y luego **Next**



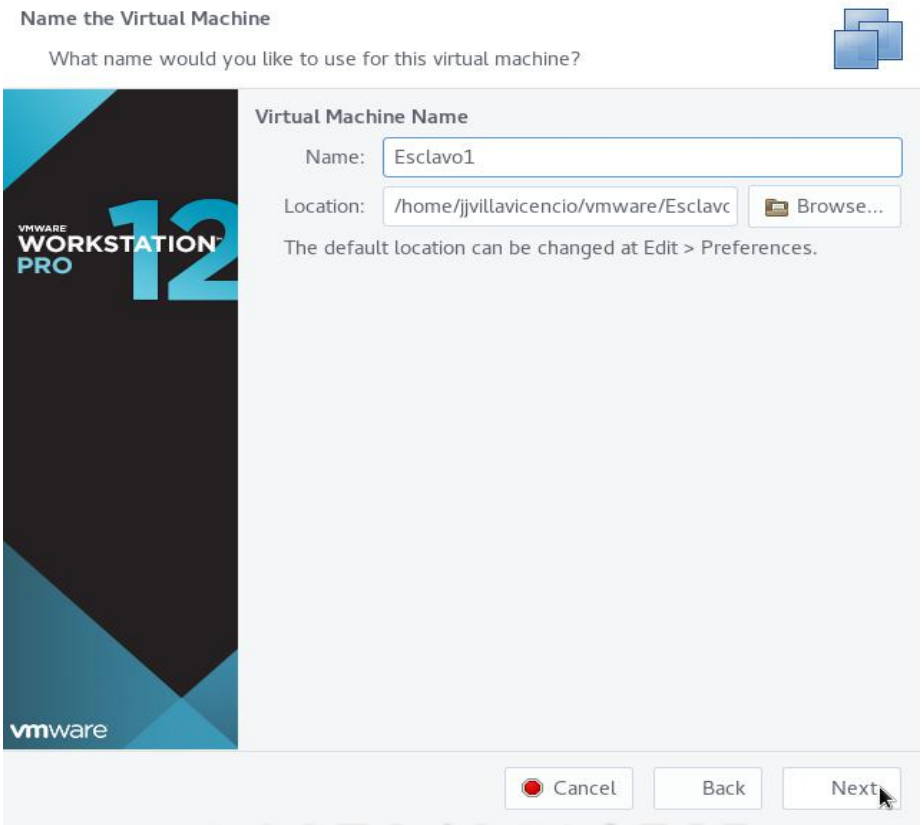
Buscamos donde tenemos nuestra **imagen (.iso) de Ubuntu 16,04** y luego **next**



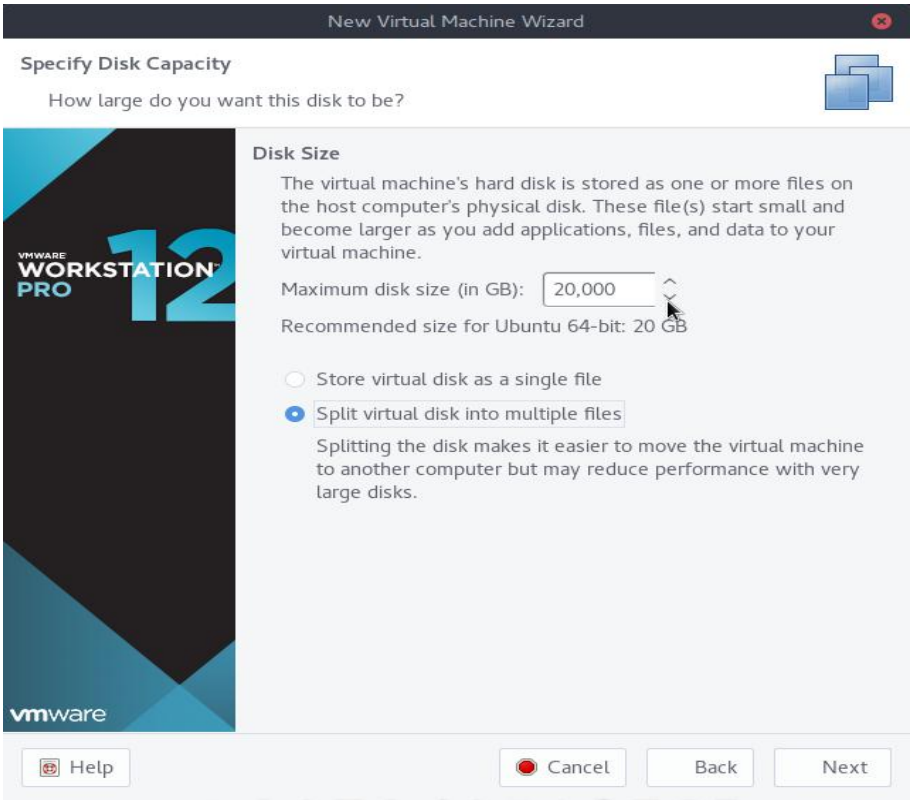
Llenamos los campos con las credenciales de acceso a nuestra distribución Ubuntu



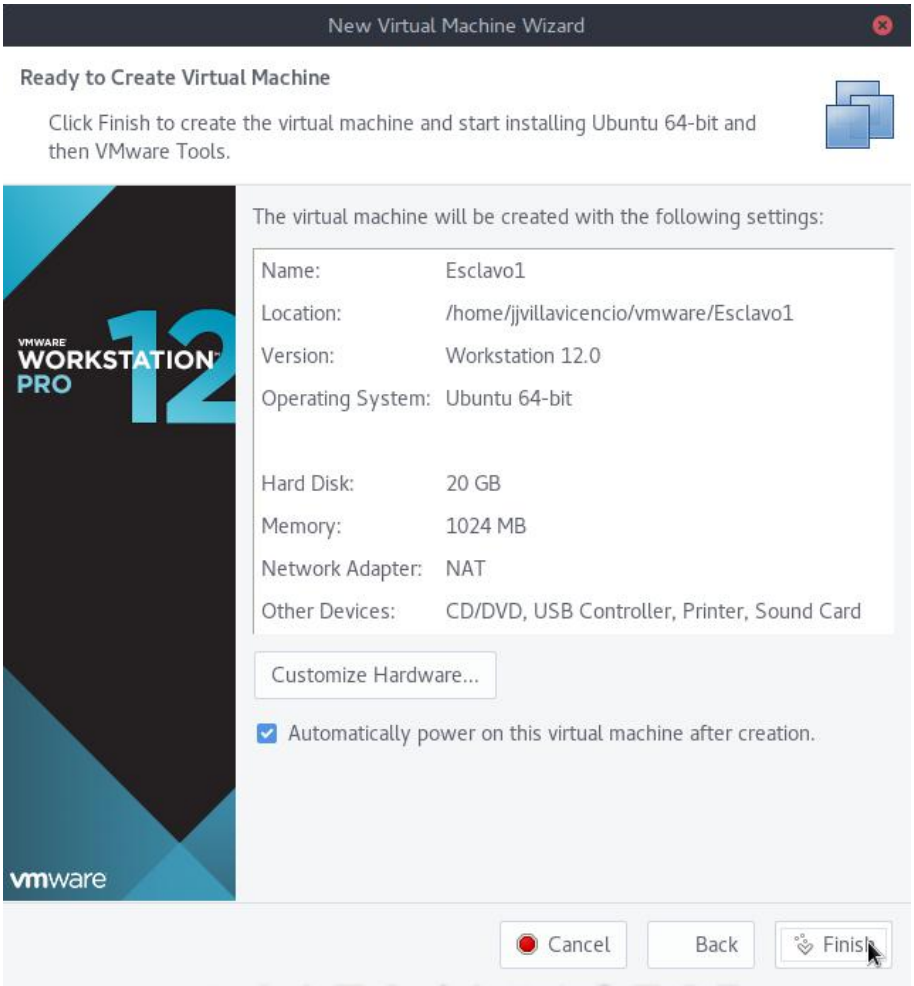
Escogemos el nombre y donde se van a guardar los datos de nuestra maquina virtual



Escogemos la cantidad que creamos necesaria de disco duro a utilizar por defecto son 20 GB.



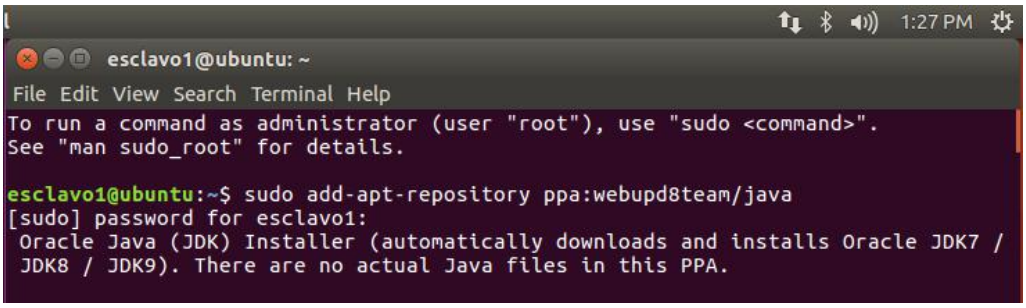
En este apartado podemos dar click en **Customize Hardware** para modificar configuraciones como Ram y cantidad de procesadores, en este caso dejaremos los valores por defecto.

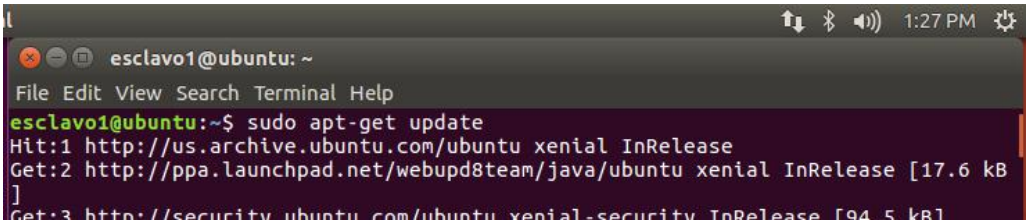


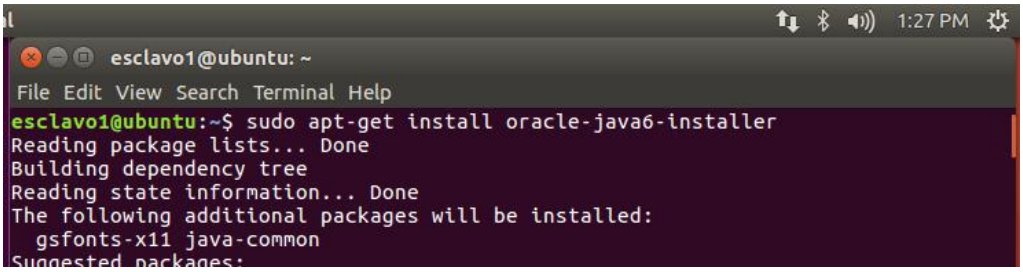
Luego iniciara el asistente de Ubuntu 16,04, y se procederá a una instalación típica de un Sistema Operativo

2. Instalación de Java SDK 6

```
# optional - remove openjdk if your installed it
sudo apt-get purge openjdk*
# install Oracle Java SDK 6
sudo add-apt-repository ppa:webupd8team/java
sudo apt-get update
sudo apt-get install oracle-java6-installer
# specify the JAVA_HOME environment variable in /etc/environment
sudo nano /etc/environment
JAVA_HOME=/usr/lib/jvm/java-6-oracle/
# force OS to reload the /etc/environment file
source /etc/environment
```







Ahora verificaremos si esta bien instalado y configurado JAVA:

```
1 java -version
2 echo $JAVA_HOME
```

```
esclavo1@ubuntu:~$ java -version
java version "1.6.0_45"
Java(TM) SE Runtime Environment (build 1.6.0_45-b06)
Java HotSpot(TM) 64-Bit Server VM (build 20.45-b01, mixed mode)
esclavo1@ubuntu:~$ echo $JAVA_HOME
/usr/lib/jvm/java-6-oracle/
esclavo1@ubuntu:~$
```

3. Instalación de Scala 2.10.4

Descargar el instalador **.deb** de aquí.

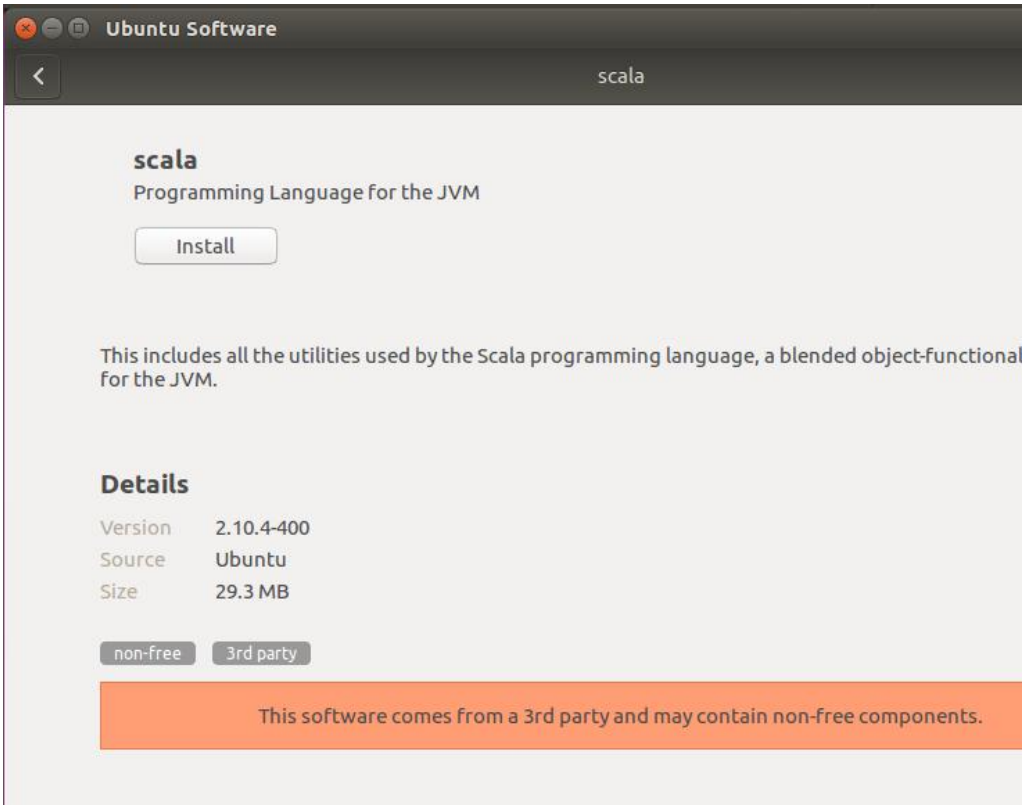
www.scala-lang.org/download/2.10.4.html

You can find the installer download links for other operating systems, as well as documentation and source code archives for Scala 2.10.4 below.

Archive	System	Size
scala-2.10.4.tgz	Mac OS X, Unix, Cygwin	28.55M
scala-2.10.4.msi	Windows (msi installer)	60.00M
scala-2.10.4.zip	Windows	28.60M
scala-2.10.4.deb	Debian	24.83M
scala-2.10.4.rpm	RPM package	24.83M
scala-docs-2.10.4.txz	API docs	3.65M
scala-docs-2.10.4.zip	API docs	32.46M
scala-sources-2.10.4.zip	sources	
scala-tool-support-2.10.4.tgz	Scala Tool Support (tgz)	25K
scala-tool-support-2.10.4.zip	Scala Tool Support (zip)	46K

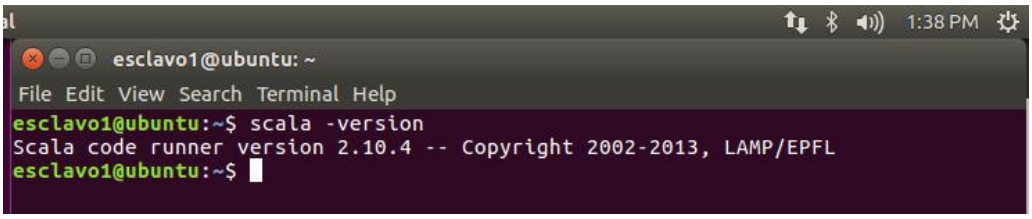
Firefox automatically sends some data to Mozilla so that we can improve your experience. Choose What I Share

Luego hacer doble clic en el archivo descargado e instalarlo con el asistente de Ubuntu.



Ahora ejecutamos en la Terminal las siguientes lineas para comprobar la instalación:

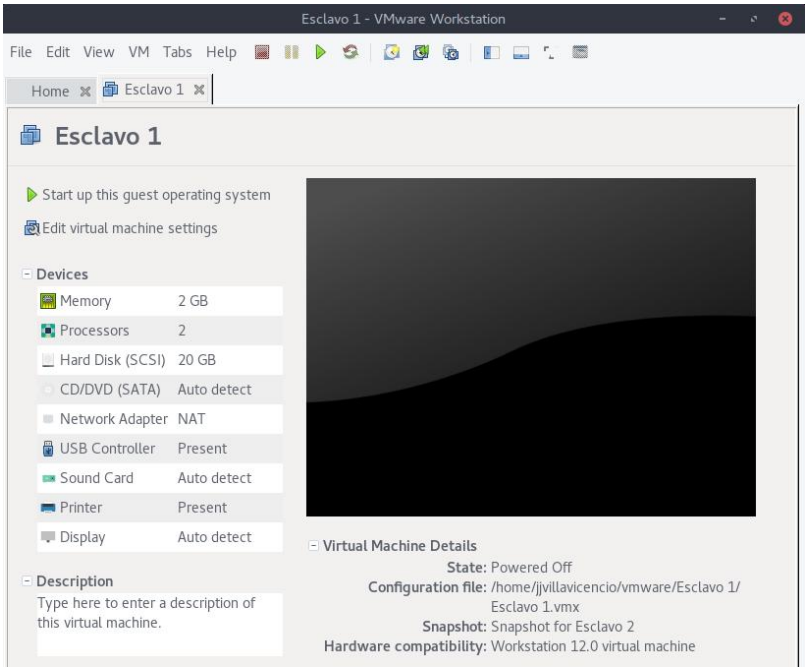
```
scala -version
```



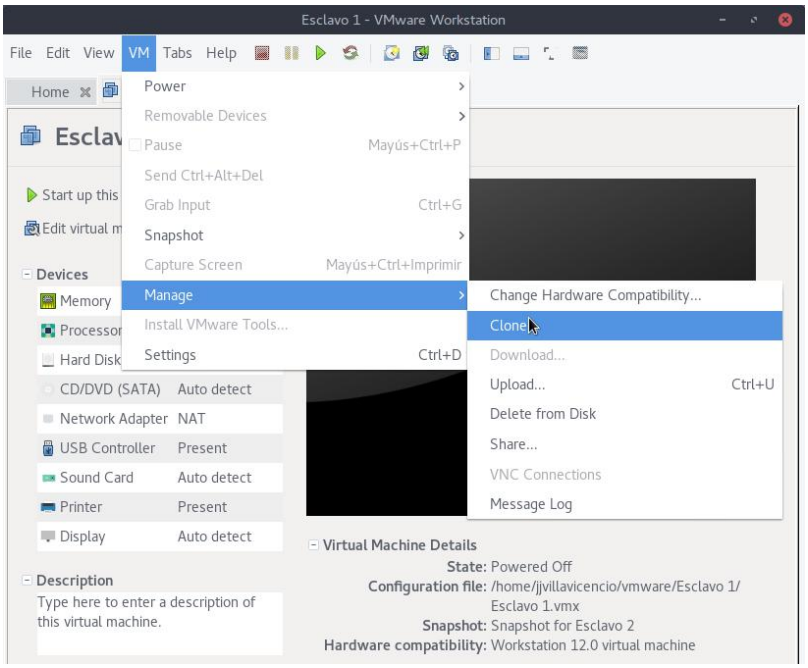
4. Clonar maquina virtual

Nota: Los siguientes pasos los repetiremos según la cantidad de nodos que deseemos en nuestro cluster.

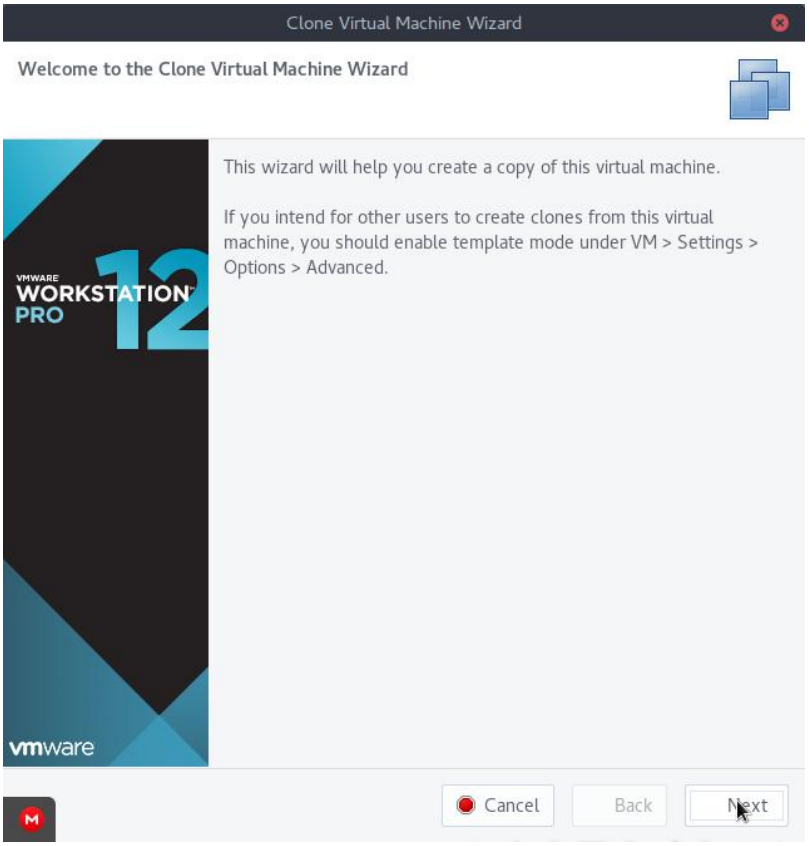
Apagamos la maquina virtual



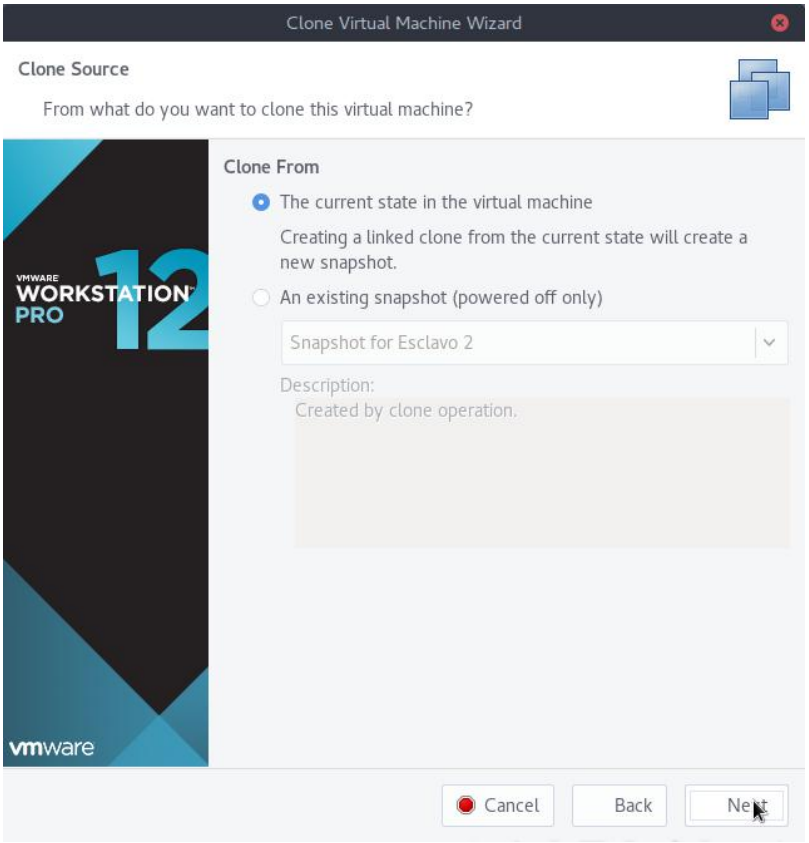
Vamos a **VM>Manage>Clone**



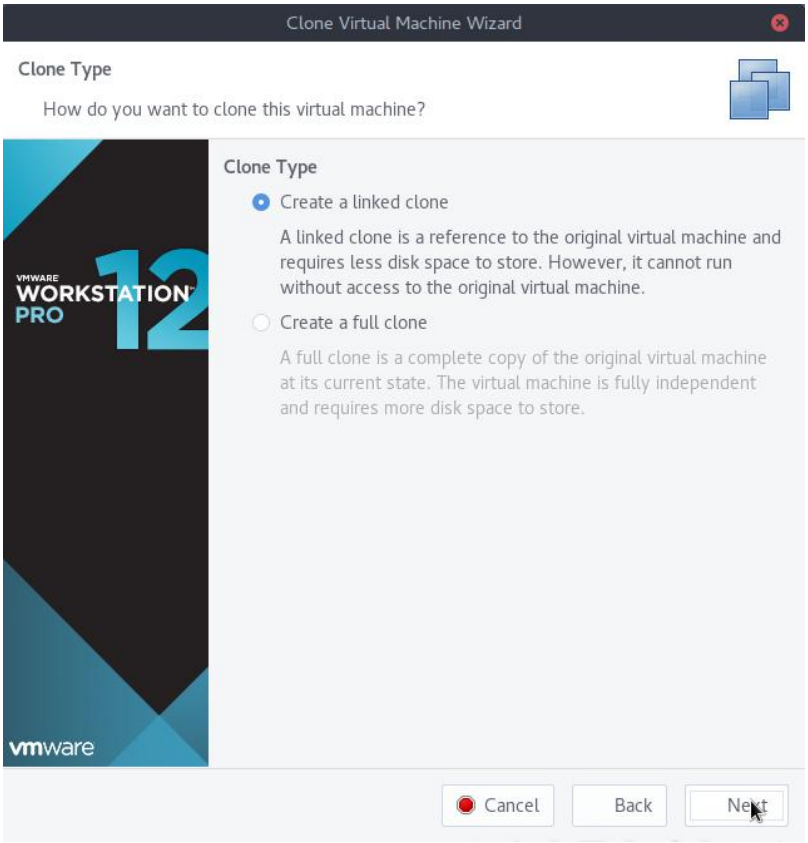
Damos siguiente en el asistente.



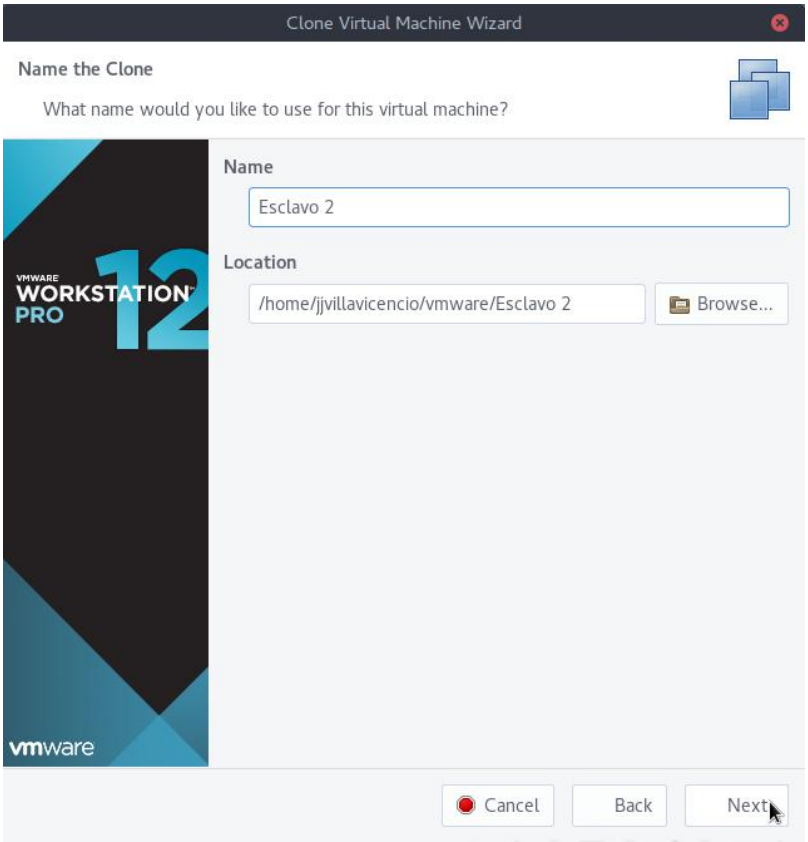
Seleccionamos Clone From The current state



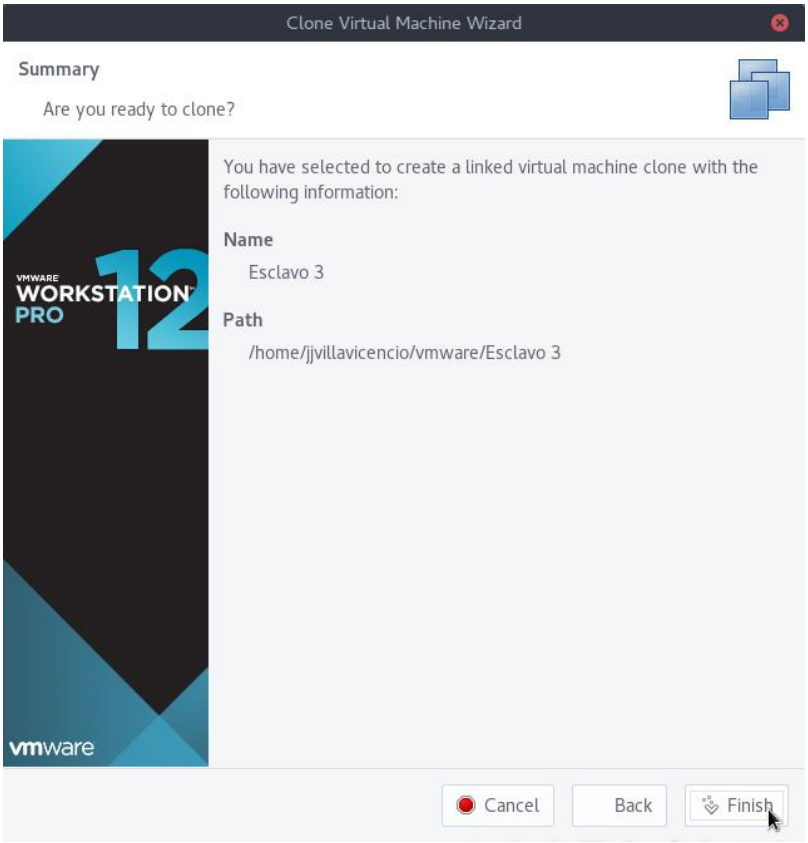
Seleccionamos **Create a linked clone**



Detallamos el nombre y donde se almacenara la nueva maquina virtual



Seleccionamos **Finish** para culminar con la clonación.



En nodo

5. Instalación SSH Acceso Remoto

Nodo Master

IP: 172.16.233.128

Usuario: esclavo1

Nodo esclavo

IP: 172.16.233.129

Usuario: esclavo1

```
# En los nodos esclavos instalamos el SSH Server para que el nodo Master acceda
sudo apt-get install openssh-server
# En el nodo Master generamos la clave rsa para acceso remoto
ssh-keygen
# Para acceder via SSH sin contraseña a los nodos esclavo, copiamos la clave
ras desde el nodo Master a los nodos esclavo (el usuario e ip son del nodo esclavo)
ssh-copy-id -i ~/.ssh/id_rsa.pub esclavo1@172.16.233.129
```

Nodo Esclavo:

```
esclavo1@ubuntu: ~
esclavo1@ubuntu:~$ sudo apt-get install openssh-server
[sudo] password for esclavo1:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-client openssh-sftp-server ssh-import-id
Suggested packages:
  ssh-askpass libpam-ssh keychain monkeysphere rssh molly-guard
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
The following packages will be upgraded:
  openssh-client
1 upgraded, 4 newly installed, 0 to remove and 246 not upgraded.
Need to get 636 kB/1,223 kB of archives.
After this operation, 5,145 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 ncurses-term all 6.0
+20160213-1ubuntu1 [249 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 openssh-sftp
-server amd64 1:7.2p2-4ubuntu2.1 [38.8 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 openssh-serv
er amd64 1:7.2p2-4ubuntu2.1 [338 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 ssh-import-id all 5.
```

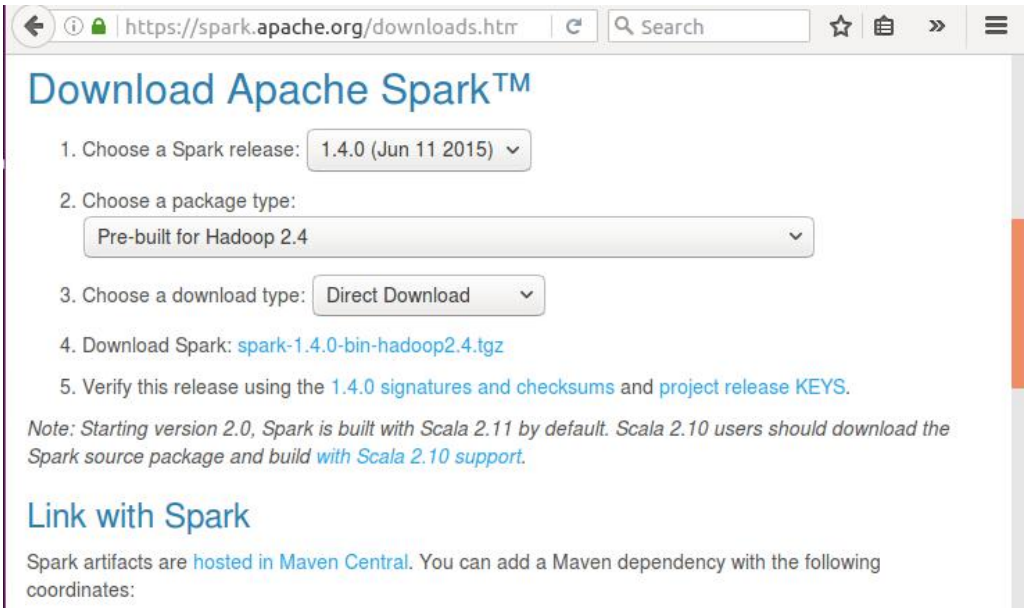
Nodo Master:

```
esclavo1@ubuntu: ~
File  Edit  View  Search  Terminal  Help
esclavo1@ubuntu:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/esclavo1/.ssh/id_rsa):
Created directory '/home/esclavo1/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/esclavo1/.ssh/id_rsa.
Your public key has been saved in /home/esclavo1/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:tH0AGLxsPBMnl5bUx46XfRjA2cnPpSRQNWq9j/kfDP8 esclavo1@ubuntu
The key's randomart image is:
+---[RSA 2048]----+
|      ..+o+.=o*o. |
|       = *.. *o*..|
|      o B. .+o=.*.|
|       B. o.o+ +.+|
|      . oS ...... |
|            .  ++ |
|               o+.|
|                .o|
|                 E|
+----[SHA256]-----+
esclavo1@ubuntu:~$
```

```
esclavo1@ubuntu: ~
Edit  View  Search  Terminal  Help
avo1@ubuntu:~$ ssh-copy-id -i ~/.ssh/id_rsa.pub jjvillavicencio@192.168.1.12

/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/esclavo1/.s
d_rsa.pub"
authenticity of host '192.168.1.124 (192.168.1.124)' can't be established.
A key fingerprint is SHA256:MXTIcV37fO8GqQm+78Wp8/lgNTq0dhGcz99ixge5URA.
you sure you want to continue connecting (yes/no)? yes
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
any that are already installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
ow it is to install the new keys
llavicencio@192.168.1.124's password:

er of key(s) added: 1

try logging into the machine, with:   "ssh 'jjvillavicencio@192.168.1.124'"
check to make sure that only the key(s) you wanted were added.
```

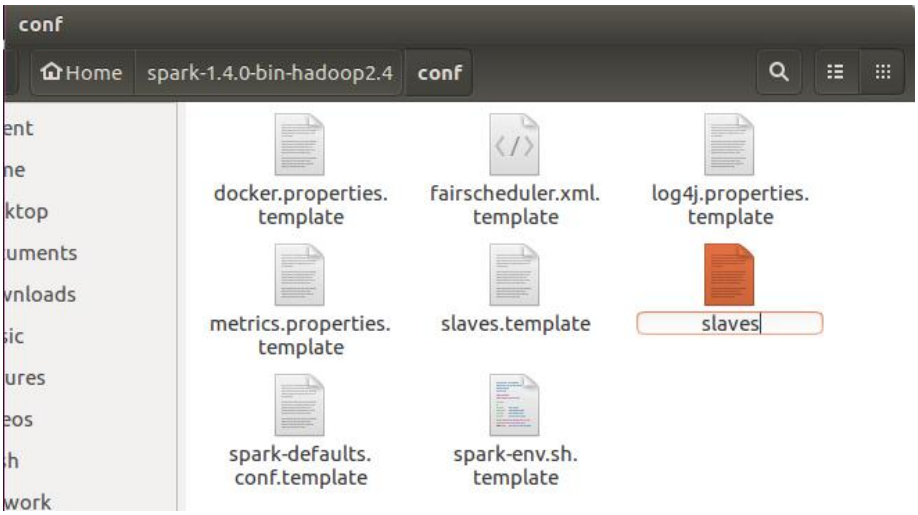
6. Instalación de Spark

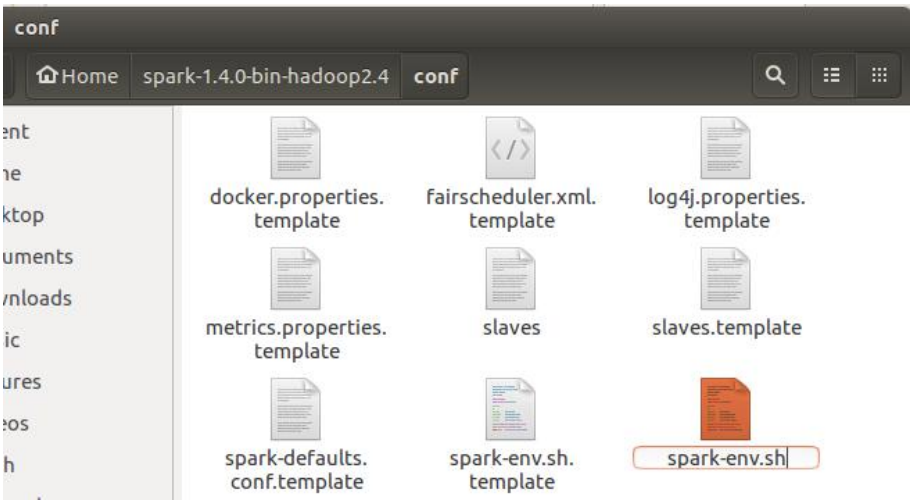
Descargamos Apache Spark 1.4.0 Prebuild for Hadoop 2.4 de aquí.



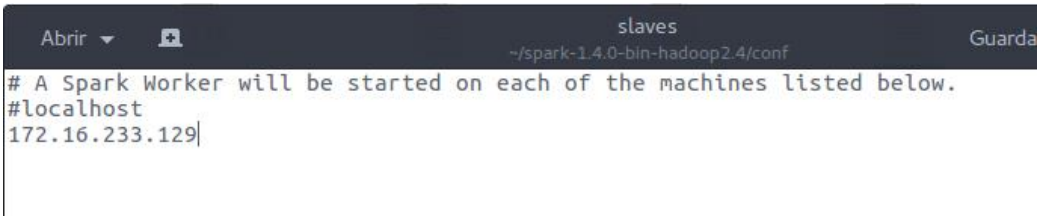
Pasos a seguir en Nodo Master y Nodo Esclavo:

- Descomprimimos el archivo **spark-1.4.0-bin-hadoop2.4** en la carpeta **Home** de nuestros Nodos
- En la carpeta **spark-1.4.0-bin-hadoop2.4/conf** renombramos los archivos **slaves.templates** y **spark-env.sh.template** a **slaves** y **spark-env.sh** respectivamente



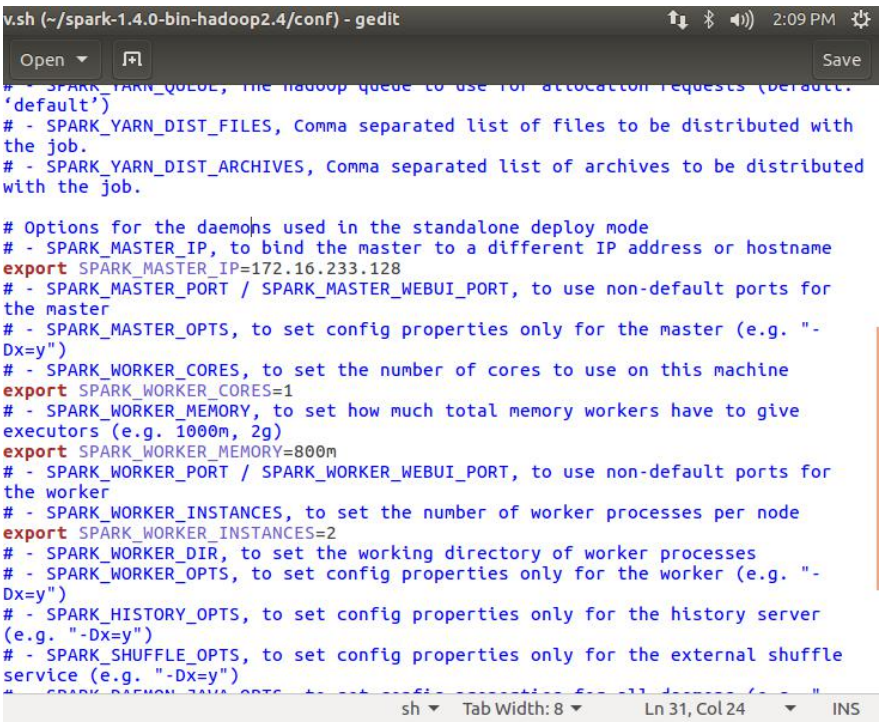


- Editamos el archivo slaves y comentamos la linea localhost y agregamos todas las IP's de los nodos esclavo



- Editamos el archivo spark-env.sh y agregamos las siguientes lineas de configuración.

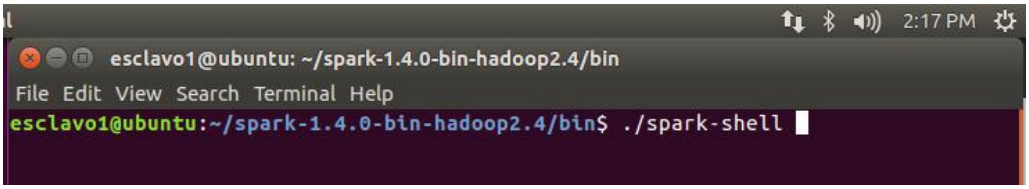
export SPARK_MASTER_IP=172.16.233.128 # IP nodo Master
export SPARK_WORKER_CORES=1
export SPARK_WORKER_MEMORY=800m
export SPARK_WORKER_INSTANCES=2

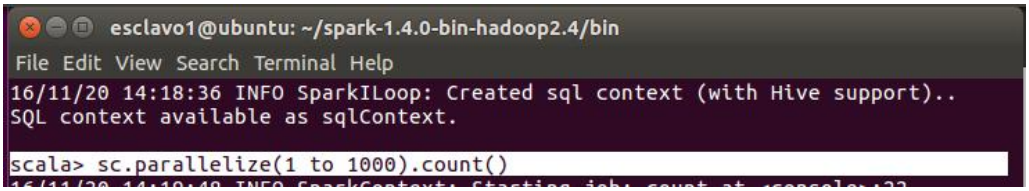


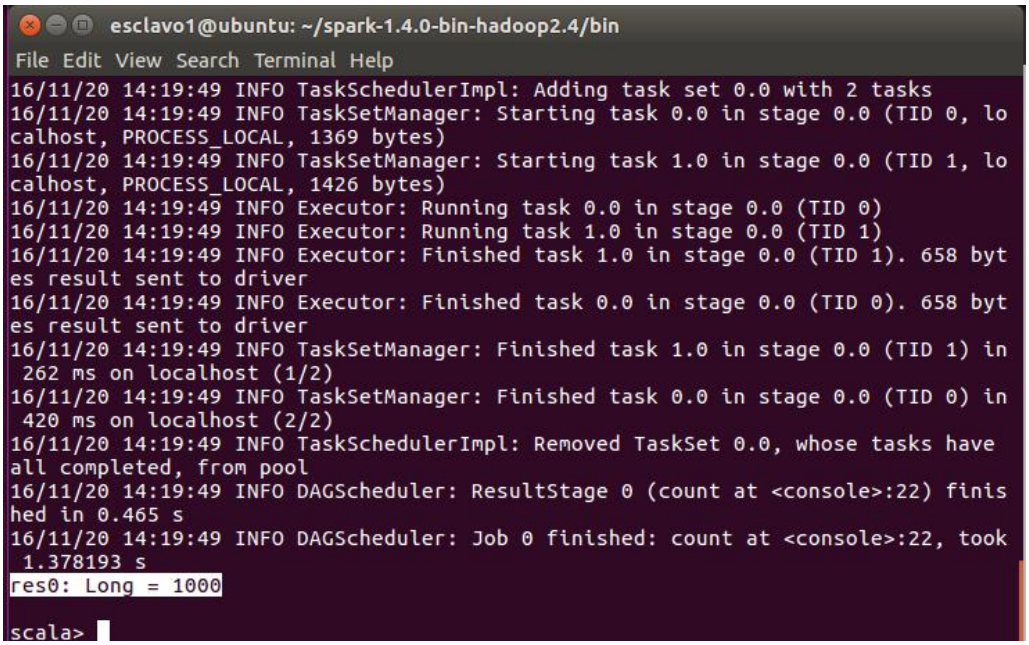
7. Pruebas

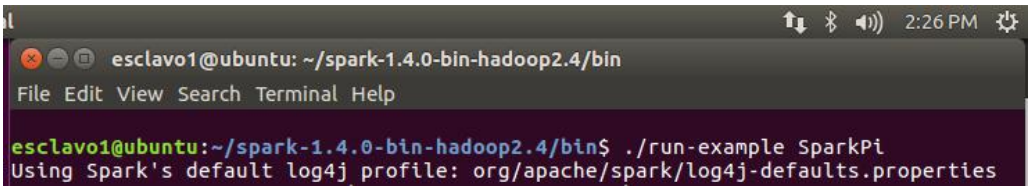
a) Probar versión local

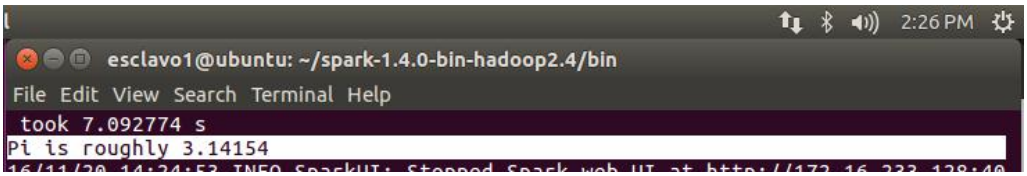
```
./bin/spark-shell #Launch the Spark shell
scala:> sc.parallelize(1 to 1000).count() # it should return 1000
scala:> exit # exit spark shell
./bin/run-example SparkPi # run the Pi example
```





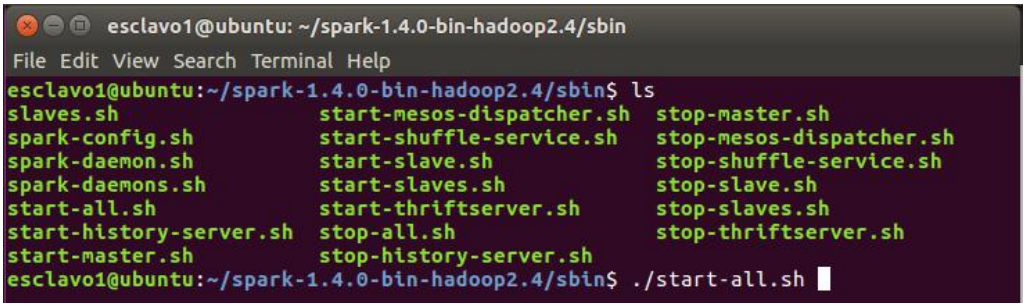




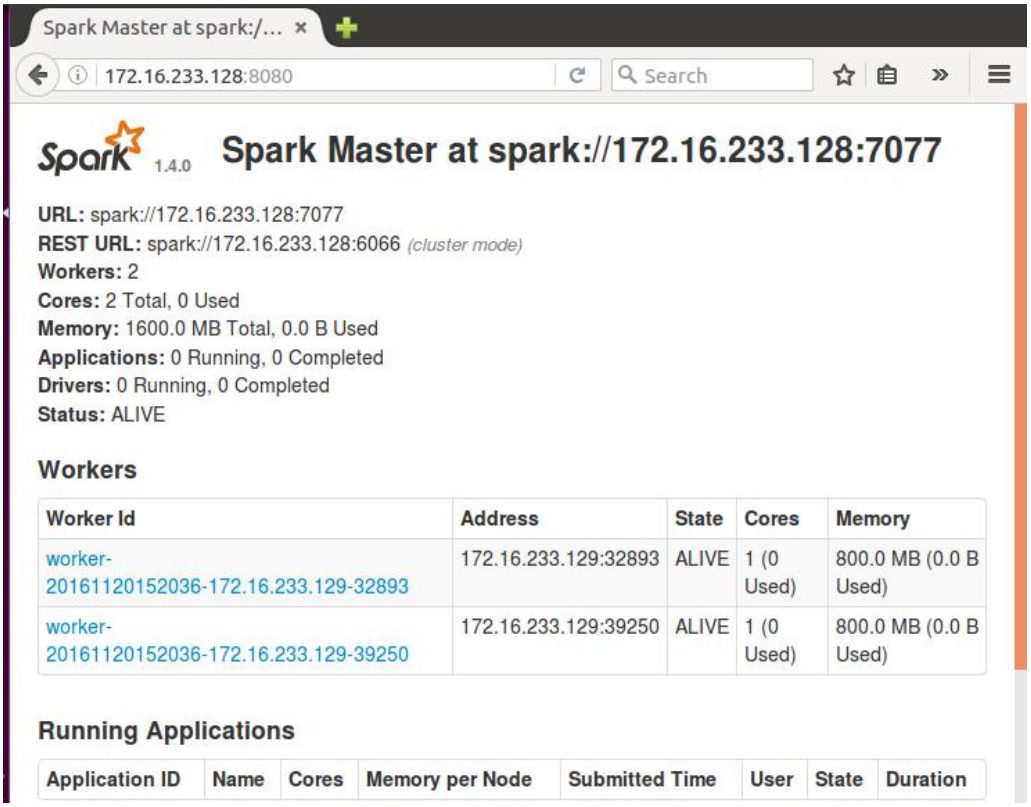


b) Iniciar y probar el cluster.

```
./sbin/start-all.sh # start our cluster
./sbin/stop-all.sh # if you want to stop our cluster
```



Ingresamos en un navegador la IP del nodo master con el puerto 8080



Ejecutamos el shell de spark que se ejecuta ya en el cluster

```
MASTER=spark://192.168.85.135:7077 ./bin/spark-shell
scala:> sc.parallelize(1 to 1000).count() # it should return 1000
```

```
esclavo1@ubuntu: ~/spark-1.4.0-bin-hadoop2.4
esclavo1@ubuntu:~/spark-1.4.0-bin-hadoop2.4$ MASTER=spark://172.16.233.128:7077
./bin/spark-shell

log4j:WARN No appenders could be found for logger (org.apache.hadoop.metrics2.li
b.MutableMetricsFactory).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more in
fo.
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
16/11/20 15:29:01 INFO SecurityManager: Changing view acls to: esclavo1
16/11/20 15:29:01 INFO SecurityManager: Changing modify acls to: esclavo1
16/11/20 15:29:01 INFO SecurityManager: SecurityManager: authentication disabled
; ui acls disabled; users with view permissions: Set(esclavo1); users with modif
y permissions: Set(esclavo1)
16/11/20 15:29:01 INFO HttpServer: Starting HTTP Server
16/11/20 15:29:01 INFO Utils: Successfully started service 'HTTP class server' o
n port 42148.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 1.4.0
      /_/
```

En el panel de control podemos ver la aplicación que se ejecuto

Spark Master at spark:/...

172.16.233.128:8080

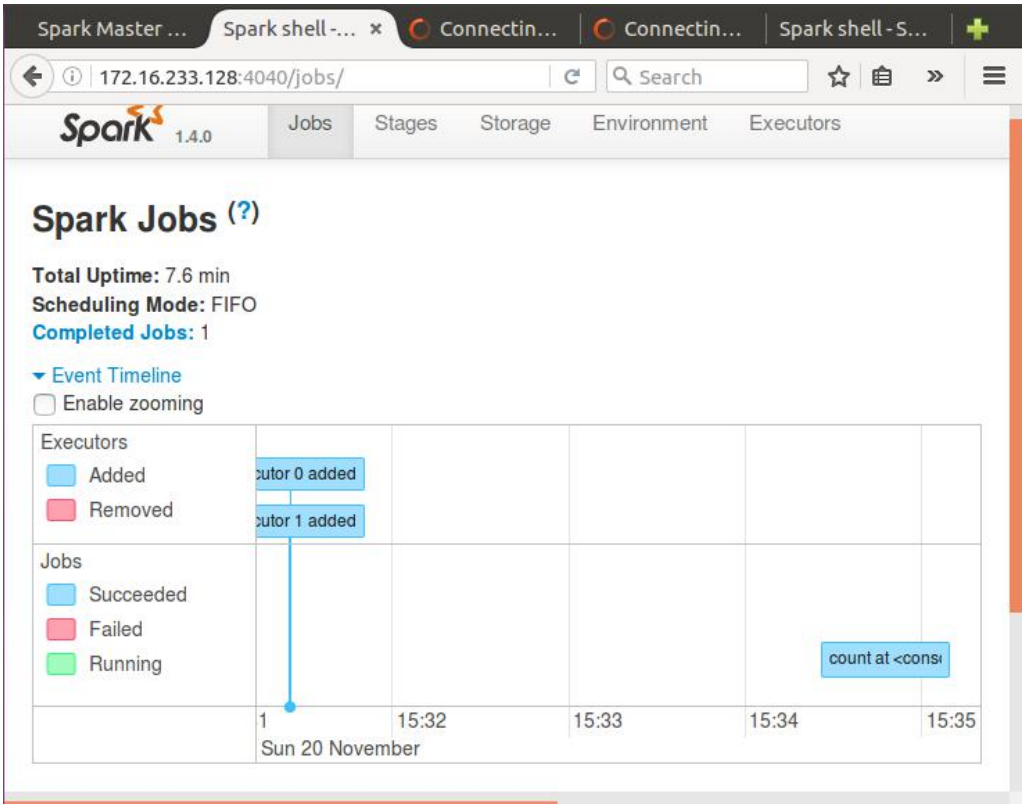
Applications: 1 Running, 0 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers

Worker Id	Address	State	Cores	Memory
worker-20161120152036-172.16.233.129-32893	172.16.233.129:32893	ALIVE	1 (1 Used)	800.0 MB (512.0 MB Used)
worker-20161120152036-172.16.233.129-39250	172.16.233.129:39250	ALIVE	1 (1 Used)	800.0 MB (512.0 MB Used)

Running Applications

Application ID	Name	Cores	Memory per Node	Submitted Time	User	State	Duration
app-20161120152945-0000 (kill)	Spark shell	2	512.0 MB	2016/11/20 15:29:45	esclavo1	RUNNING	3.0 min



c) Detener el cluster

```
./sbin/stop-all.sh # if you want to stop our cluster
```