# VERSION INFORMATION

These are the versions used for this document.

|  |  |
| --- | --- |
| Vagrant | 1.4.3 |
| VirtualBox | 4.3.6-91406-Win |
| XData VM | 0.1 |
| Putty | development snapshot 2010-12-01:r9020 |
|  |  |

# DOCUMENT CONVENTIONS

***{}*** will be used to denote optional instructions

### TODO will be used to denote areas of the install that need fixed

[ ] will be used for descriptive commands to be executed as the user sees fit

> will be used for precise commands to be executed by the user

*“ “* will be used for notes about a specific command, usually occurring after the command it notes on

# GET THE VM

* [Install Vagrant: http://www.vagrantup.com]
* [Install Virtual Box: https://www.virtualbox.org/wiki/Downloads]
* [Download VM (xdata-0.1.box) from https://drive.google.com/uc?id=0B54T370AV5JDUEF3T21qWXRpeEU&export=download (Use IE / Firefox. Chrome fails at the end of the download)]

### Load VM in Vagrant

1. > vagrant box add xdata-vm-[version] xdata-[version].box
2. [create "VM\_Home" directory to store local VM files (ex: C:\Users\<your home directory\Documents\VMs\xdata-vm-[version]\)]
3. > cd "VM\_Home" directory
4. > vagrant init xdata-vm-[version]

***{ Optional Modifications to Vagrant config file "Vagrantfile" }***

1. [edit Vagrantfile in "VM\_Home]
2. [uncomment the following line: # config.vm.network :public\_network]

*“this allows the VM to be assigned a public interface so you can access it from your Host PC*

*the alternative is to map VM ports to Host ports and access them via localhost”*

1. [replace the following lines:]

*# config.vm.provider :virtualbox do |vb|*

*# # Don't boot with headless mode*

*# vb.gui = true*

*#*

*# # Use VBoxManage to customize the VM. For example to change memory:*

*# vb.customize ["modifyvm", :id, "--memory", "1024"]*

*# end*

[with these lines:]

*config.vm.provider :virtualbox do |vb|*

*# # Don't boot with headless mode*

*# vb.gui = true*

*#*

*# # Use VBoxManage to customize the VM. For example to change memory:*

*# vb.customize ["modifyvm", :id, "--memory", "1024"]*

*vb.customize ["modifyvm", :id, "--ioapic", "on"]*

*vb.customize ["modifyvm", :id, "--cpus", "8"]*

*end*

*“Important note: the value "8" represents the number of cpus/cores from the host machine that*

*you want to grant access to the VM. It does not have to be 8 and should not exceed the available*

*cpus/cores on the host machine”*

***{ End Optional }***

1. > vagrant up

*“ this command starts the VM on the host machine ”*

# CONNECT TO THE VM

1. [Open up putty and create a connection to localhost:2222 with username: bigdata password: bigdata]

***{ Optional commands if your customized your Vagrantfile as recommended above }***

1. > sudo su

*“switches you from the user vagrant to the user root”*

1. > lscpu

*“ this should verify that you have the appropriate number of cpus available to the VM”*

1. > ifconfig

*“this should verify that you have the appropriate network interfaces up and running.*

*Interface 'eth1' should be present and contain an IP reachable by your host.*

*You should note the public interface's IP address for later”*

1. [On the host computer, create a 'host' file entry for the public IP address of the VM (ex. 172.16.10.13 xdata)]
2. [On the VM remove all references of xdata to the localhost IP (127.0.0.1) and create references from xdata to the public ip address identified above]
3. > ping xdata

*“this should now ping the public interface of your VM”*

1. [restart the VM]
2. > vagrant halt
3. > vagrant up

*“this will bring up some of the hbase services on the correct interface”*

1. [SSH back into the VM as bigdata]

***{ End Optional }***

1. [create the following browser bookmarks to the VM

http://xdata:50070/dfshealth.jsp

http://xdata:50090/status.jsp

http://xdata:50030/jobtracker.jsp

http://xdata:50060/tasktracker.jsp

http://xdata:9095/thrift.jsp

http://xdata:60010/master-status

http://xdata:60030/rs-status

http://xdata:8085/rest.jsp

]

*“ replace xdata with localhost if you didn't create a public interface for your VM*

*if you use localhost then it will be necessary to map those ports from Host to VM”*

# INSTALL LOUVAIN-MODULARITY

1. > wget https://github.com/Sotera/distributed-louvain-modularity/releases/download/v1.0\_chd4.5.0\_giraph1.0.0/distributed-louvain-modularity-cdh4.5.0-giraph1.0.0-bin.tar.gz
2. > mkdir ~/xdata
3. > tar zxf distributed-louvain-modularity-cdh4.5.0-giraph1.0.0-bin.tar.gz -C ~/xdata

# INSTALL AGGREGATE-MICRO-PATHS

1. > cd ~/xdata
2. > git clone https://github.com/Sotera/aggregate-micro-paths.git

# INSTALL TRACK-COMMUNITIES

1. > git clone https://github.com/Sotera/track-communities.git

# RUN EXAMPLES

1. > cd ~/xdata/aggregate-micro-paths/hive-streaming
2. > ./run\_ais.sh

*“this will ensure the correct tables exist for the track-communities project”*

1. > cd ~/xdata/distributed-louvain-modularity/example
2. > ./run\_example.sh

*“this will ensure the louvain-modularity is working correctly”*

1. > cd ~/xdata/track-communities
2. > chmod 744 example.sh
3. > chmod 744 run.sh
4. > ./example.sh

*“this will execute the track-community example on the ais data”*

# INSTALL TANGELO

1. > sudo apt-get install python-pip python-dev build-essential

*“this installs pythons package manager”*

1. > sudo pip install tangelo
2. > echo -e '#!'"/bin/bash\ntangelo --hostname xdata --port 8000 --verbose -nc start" > tangelo\_start.sh
3. > echo -e '#!'"/bin/bash\ntangelo stop" > tangelo\_stop.sh
4. > chmod 744 tangelo\_s\*.sh

*“the above two create a start and stop script for tangelo.”*

1. > cd ~/
2. > ln -s ~/xdata/track-communities/tangelo\_html/ tangelo\_html

“creates a symbolic link in the home directory for the tangelo\_html directory”

### TODO the current version of 'tangelo\_html' references files that don't exist in the latest version of Tangelo

1. [edit ~/tangelo\_html/myservice.py and replace 'response = tangelo.empty\_response()' with 'response = {}']

### TODO END ###

1. [edit ~/tangelo\_html/paths.js and replace 'localhost' with 'xdata']
2. [edit ~/tangelo\_html/myservice.py and replace 'localhost' with 'xdata' and replace 'FILL\_THIS\_IN' with 'ais\_small\_final\_tracks\_comms\_joined']
3. > cd xdata/track-communities/commserver/
4. > sudo python setup.py install
5. > cd ~/
6. > ln -s xdata/track-communities/commserver/go.py go.py
7. [edit ~/go.py and replace 'FILL\_THIS\_IN' with 'ais\_small\_final\_tracks\_comms\_joined' and replace 'arcus7:21000' with 'xdata:21000']

# INSTALL IMPYLA

1. > sudo pip install impyla

# START SERVICES

1. > cd /etc/init.d
2. > sudo ./impala-server start
3. > sudo ./impala-state-store start
4. > sudo ./impala-catalog start

*“starts the impala services”*

1. > cd ~/
2. >./tangelo\_start.sh

*“start the tangelo service”*

1. > sudo python go.py &

*“starts the commserver”*

# TEST SERVICES

1. [access the url 'http://xdata:8787/setcomm/244690808']ifcon
2. [access the url 'http://xdata:8000/~bigdata/']
3. [select the "Refresh" button, you should see track data]
4. [select the "Play" button, you should see track progress]