

Metal as a Service (MaaS)

Introduction:

Metal-as-a-Service, or MAAS, is a provisioning construct created by Canonical, developers of the Ubuntu Linux-based operating system. MAAS is designed to help facilitate and automate the deployment and dynamic provisioning of hyper scale computing environments such as big data workloads and cloud services. It lets you treat physical servers like virtual machines in the cloud. Rather than having to manage each server individually, MAAS turns your bare metal into an elastic cloud-like resource.

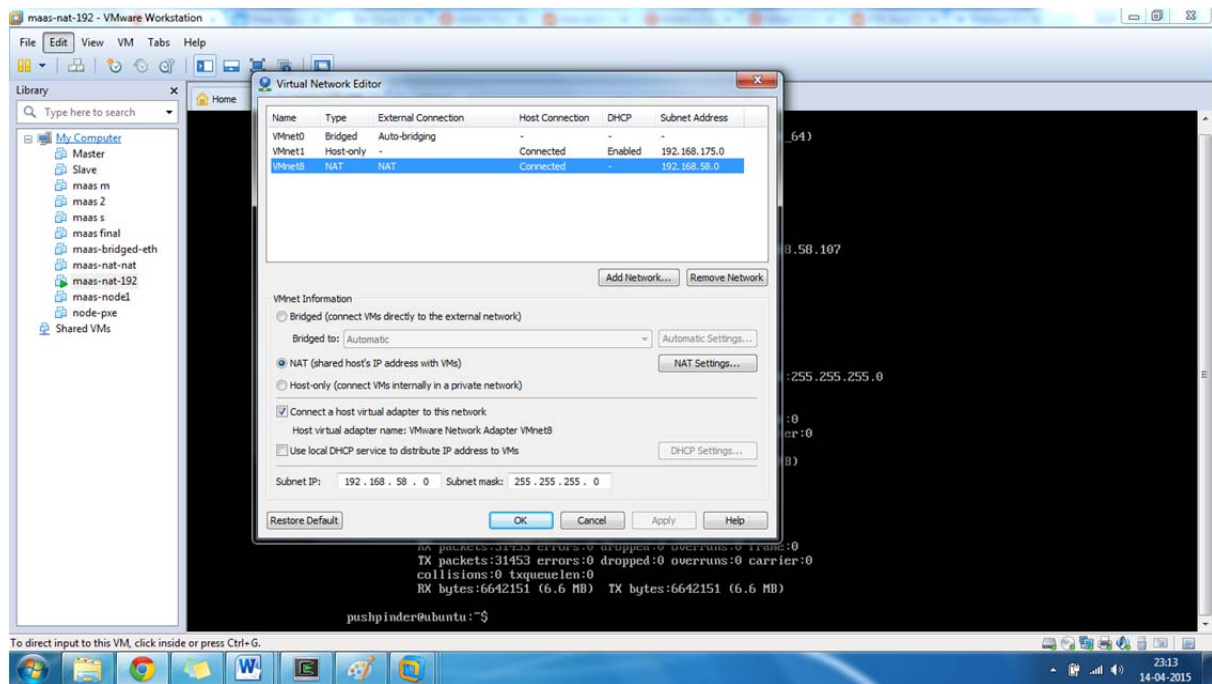
MAAS serves as a layer underneath Infrastructure-as-a-Service (IaaS) and works with Juju to coordinate applications and workloads, deploying hardware and services that can dynamically scale up and down – a cloud being just one example. It lets you provision servers dynamically, just like cloud instances – only in this case, they're whole physical nodes. With a simple web interface, you can add, commission, update and recycle your servers at will. As your needs change, you can respond rapidly, by adding new nodes and dynamically re-deploying them between services. When the time comes, nodes can be retired for use outside the MAAS. MAAS is under the AGPLv3 and requires at least Ubuntu 04.12 Beta 2.

Setting up MaaS with virtual machines:

VMware Workstation was used to create the virtual environment used to host maas. Ubuntu server 14.04 was used as guest OS.

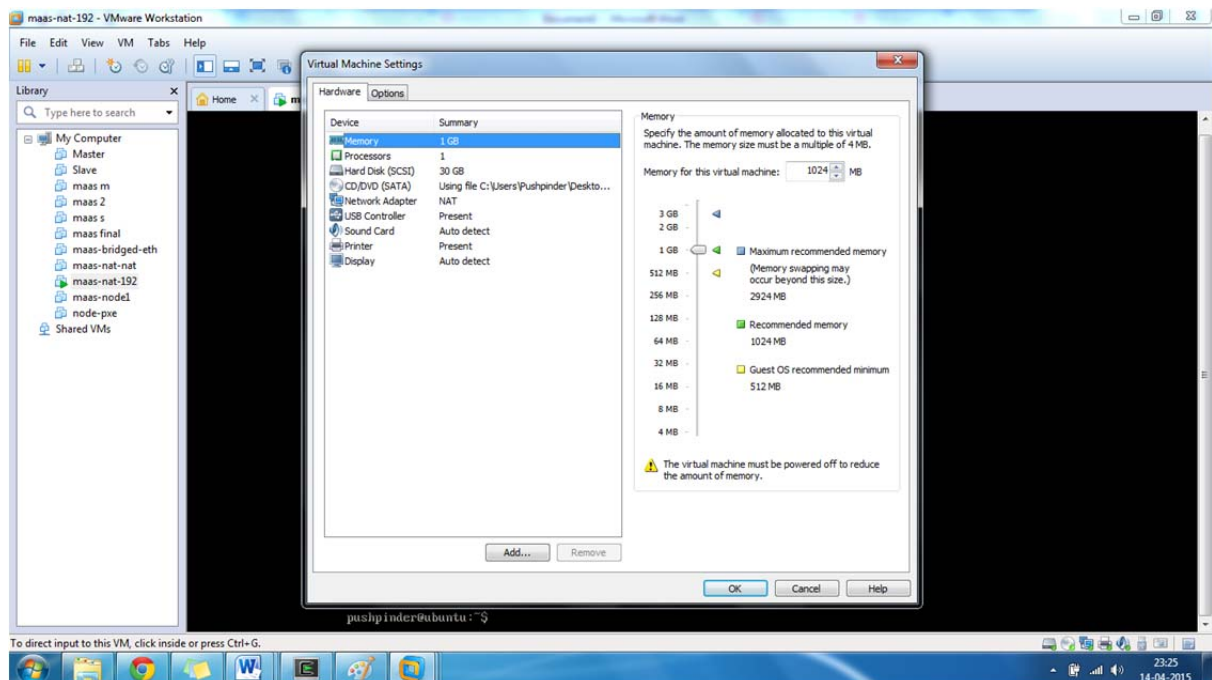
Network configuration:

One of the most challenging aspects of creating maas in a virtual environment is to get the networking correctly working as the guest machine (VM) should be able to communicate with the host machine as well as with the rest of the internet. After a lot of trial and error it was found that the main network adapter on the guest machine should use NAT and not Bridged or Host only networks. Also DHCP should be disabled as maas will create its own DHCP server to enlist nodes later on. The subnet address given to NAT should not conflict with the physical network connected to the host machine and it should not be used by any other interface. In my case, I used the subnet 192.168.58.0 /24 for NAT. This means that during the configuration of the Ubuntu server VM, the VM should be manually given an address which falls in this subnet. I chose 192.168.58.107 as the address of the VM. The following screenshot clarifies the network configuration.



VM Creation:

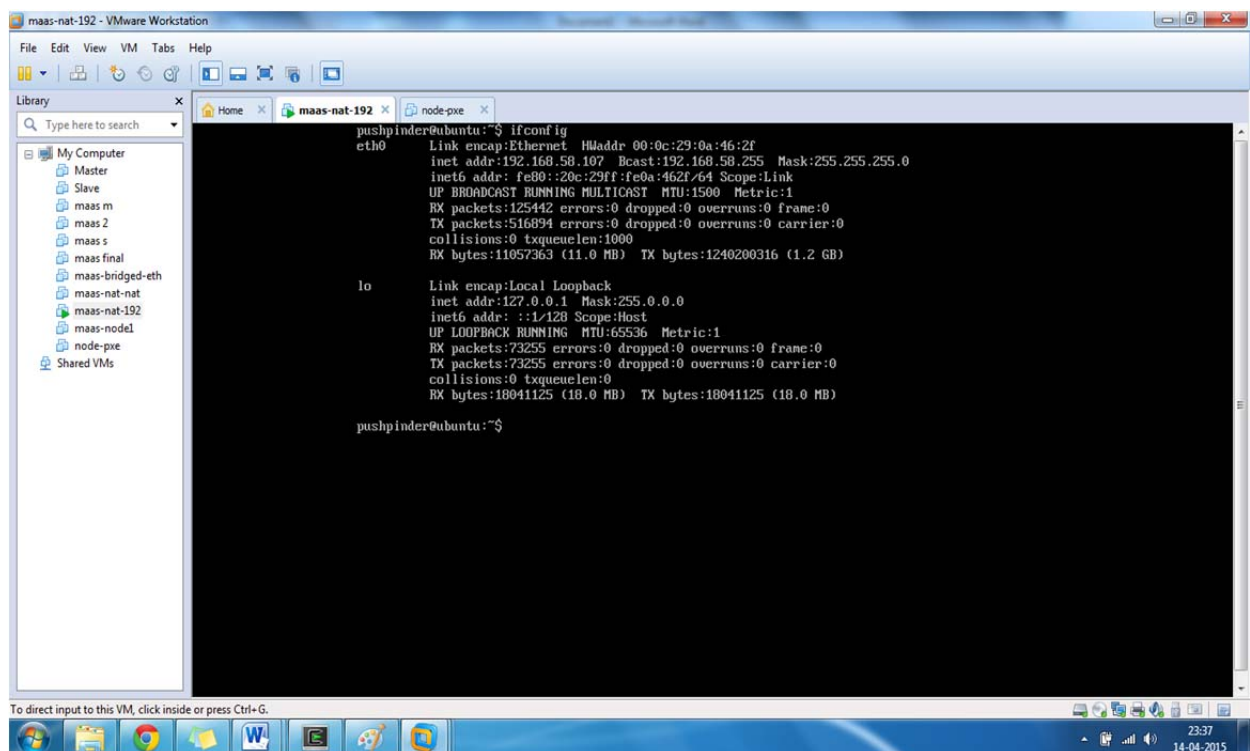
A VM was created with 1 GB RAM and 30 GB hard disk and was given one network interface connected to NAT.



Installation of Ubuntu server:

Ubuntu server 14.04 was loaded in VM using its ISO build. Single server installation (first option) was chosen instead of the “Multiserver install with MAAS” option because the latter option somehow did not allow boot images to be imported later on. After configuring the region, language, keyboard layout the installer will check the VM’s network connectivity and the DHCP configuration will fail as DHCP has been disabled so the network needs to be manually configured. For this installation, the IP was 192.168.58.107, the subnet was 255.255.255.0, the default gateway was 192.168.58.2 (click on “NAT Settings” button in first screenshot to find out), and the DNS server was 192.168.58.1. If the configuration is correct the installer should be able to get the current time and region of our system using NTP. The installation moves on and user account is created, domain name is left empty and no HTTP proxy is configured. If the system has working internet then apt will install necessary packages and the installation will complete shortly.

After the installation is complete, log into the newly created VM and check its network configuration using ifconfig. Also try sudo apt-get update to update the package list, this should also tell you about the network connectivity of the system. The output should show the IP you entered during manual network configuration during the setup.



```
maas-nat-192 - VMware Workstation
File Edit View VM Tabs Help
Library
Type here to search
My Computer
Master
Slave
maas m
maas 2
maas s
maas final
maas-bridged-eth
maas-nat-nat
maas-nat-192
maas-nodel
node-pxe
Shared VMs

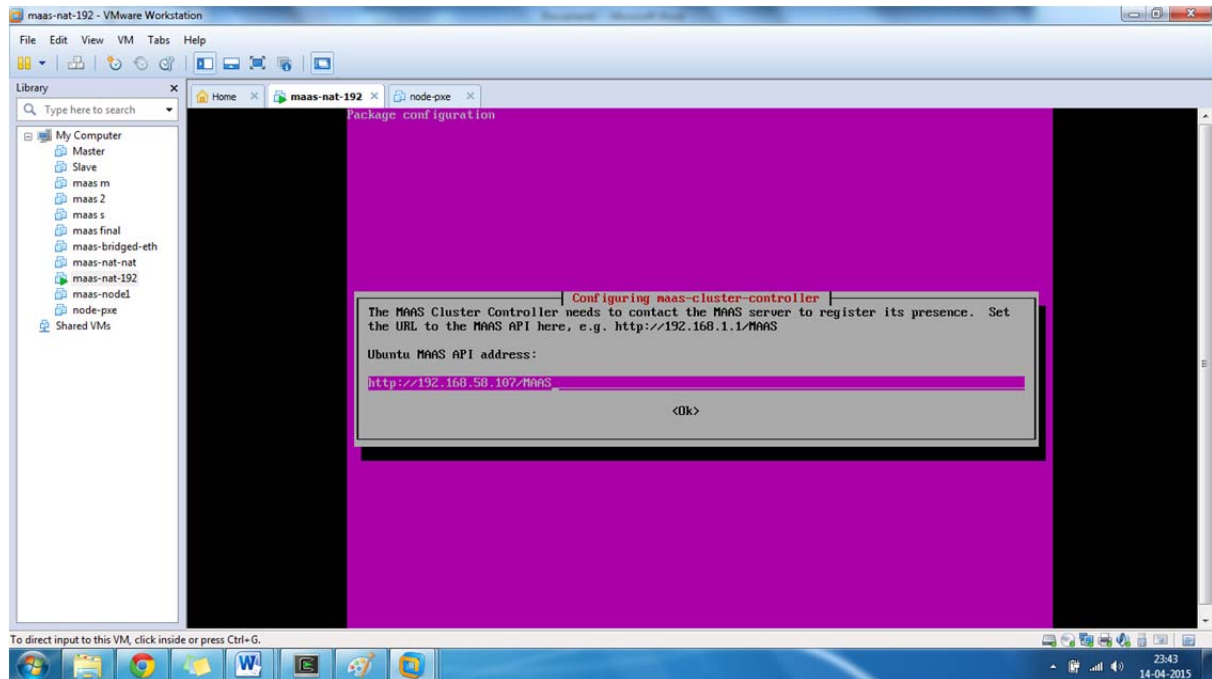
pushpinder@ubuntu:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:0a:46:2f
          inet addr:192.168.58.107  Bcast:192.168.58.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe0a:462f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:125442 errors:0 dropped:0 overruns:0 frame:0
          TX packets:516894 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:11057363 (11.0 MB)  TX bytes:1240200316 (1.2 GB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:73255 errors:0 dropped:0 overruns:0 frame:0
          TX packets:73255 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:18041125 (18.0 MB)  TX bytes:18041125 (18.0 MB)

pushpinder@ubuntu:~$
```

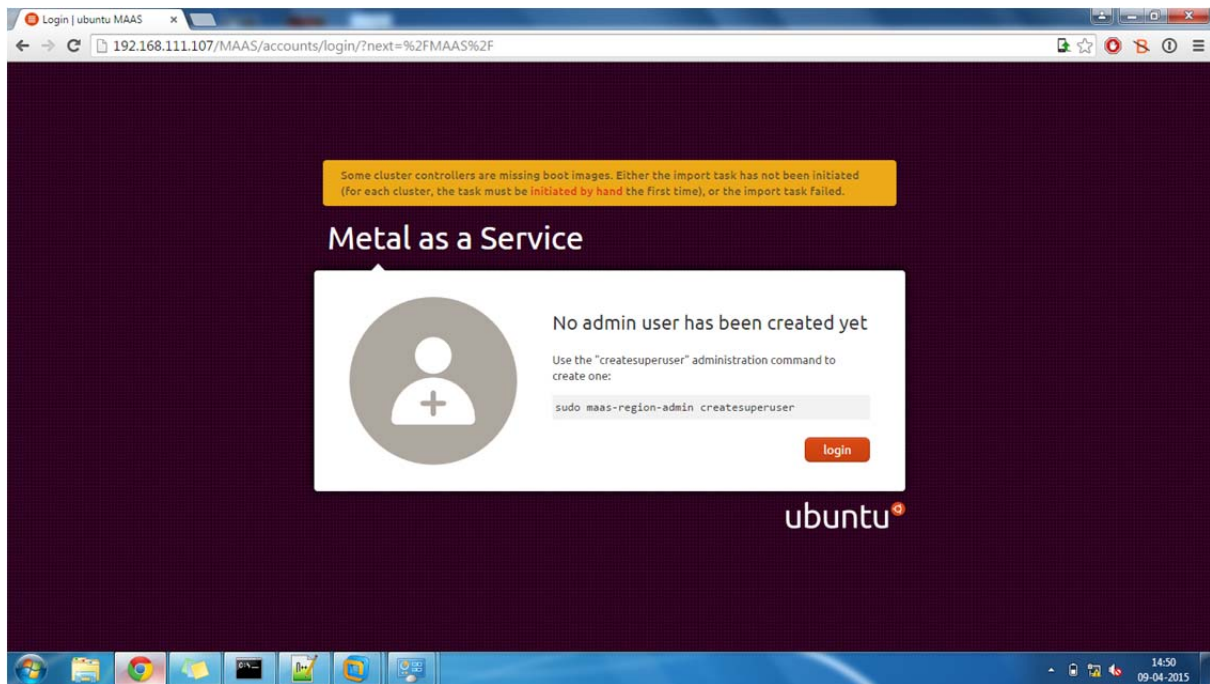
Installing Maas:

Use the command “`sudo apt-get install maas`”. This will install the required packages for maas and will require about 252 MB’s of download and depending on the network speed it may take some time. Once the download and installation is complete you can use `sudo dpkg-reconfigure maas-cluster-controller` to check the address through which maas user interface is available.



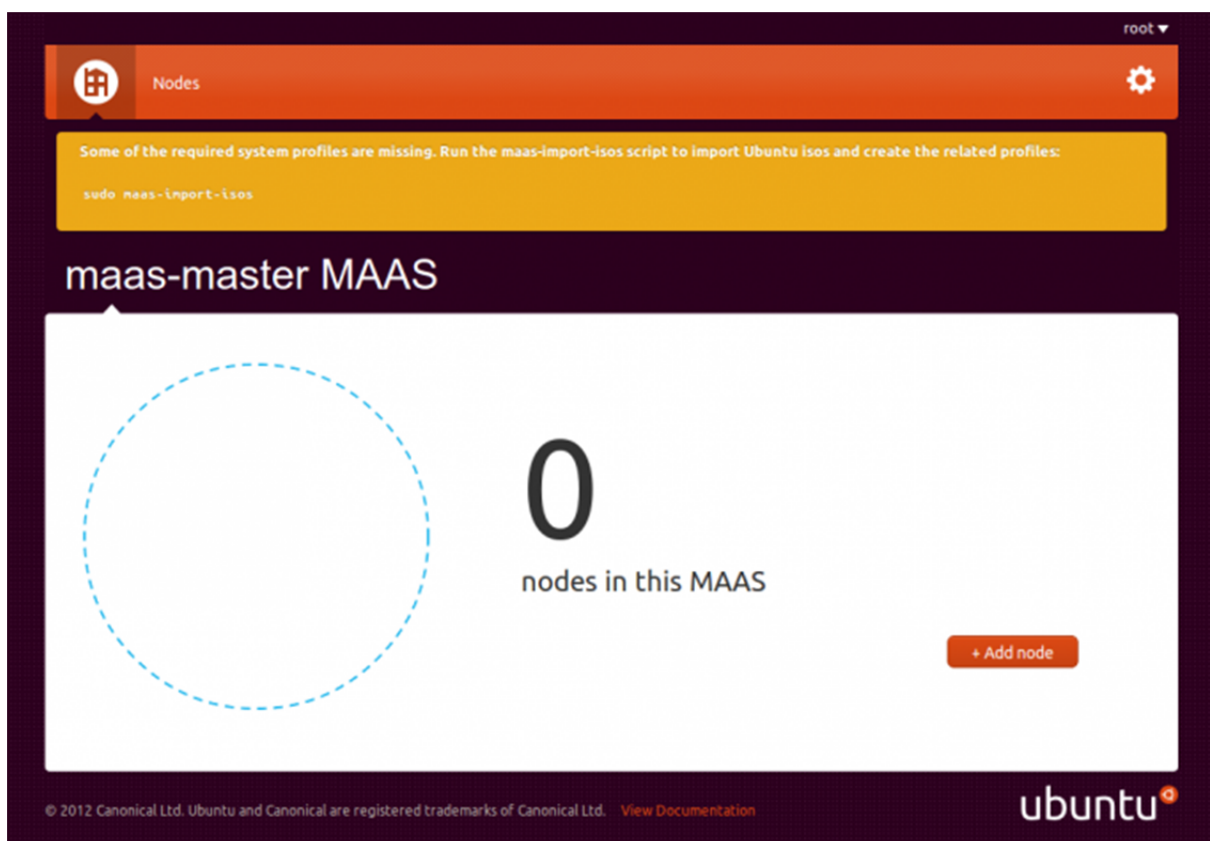
The output should have the IP of your machine followed by MAAS as the URL. You can change the address here but make sure that the new address fall in the same network range and run the command `sudo dpkg-reconfigure maas` afterwards to apply the changes. If the network is correctly set up then you can access the user interface by putting this URL in the browser on the host machine. You should get a screen like in the next screen shot.

NOTE: the following screen has a different URL then the one shown before as this screen shot was taken earlier (as can be seen by time and date in the lower right hand corner) and on a different network. Barring the URL, the output would be exactly the same.



Type the given command in the Ubuntu server VM to create a new super user and assign it a password. Leaving the username blank will automatically change the username to be root. After creating the user account in the VM you can now log in through the browser by clicking login and entering the credentials you just created.

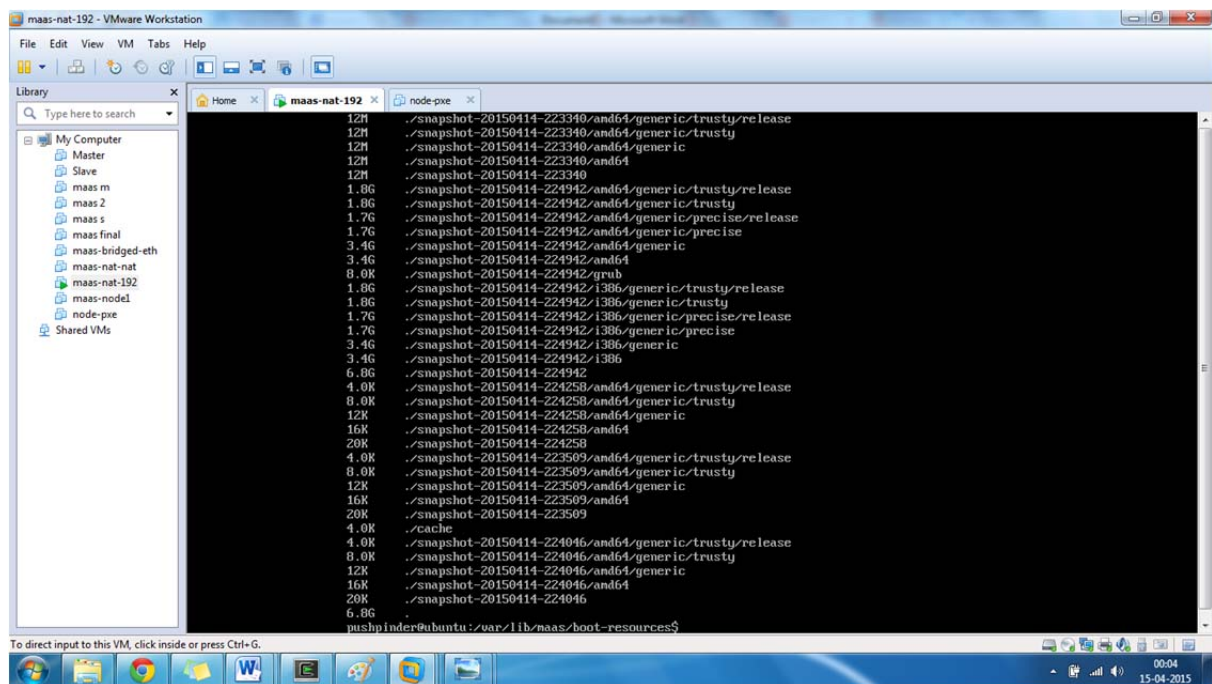
NOTE: This screen shot has hostname master but later screenshots will use Ubuntu MAAS



Importing boot images:

Before moving further and adding nodes, boot images need to be manually imported from their sources. This can be done through the user interface by clicking on “Clusters” on the home page and then clicking “Import boot images”. There will be no progress bar or any other indicator to show if the images are being downloaded. You may check your systems resource monitor to see network activity of the download.

You may also import the images using the CLI (preferred) by using “sudo maas-import-pxe-files”. Documents also mention some other commands but this one worked most reliably. Still it will not show any progress bar so you can use “ps -ae | grep maas”, if the images are being downloaded then this command will return the process maas-pxe. Another way to check the download would be to navigate to ‘/var/lib/maas/boot-resources’ and use the command “du -h” to check the directory size. The download size is 6.8 GB’s so it can take a considerably long time. On completion, the output of “du -h” should resemble:



```
maas-nat-192 - VMware Workstation
File Edit View VM Tabs Help
Library
Type here to search
My Computer
  Master
  Slave
  maas m
  maas 2
  maas final
  maas-bridged-eth
  maas-nat-nat
  maas-nat-192
  maas-nodel
  node-pxe
  Shared VMs
maas-nat-192
node-pxe
12M  ./snapshot-20150414-223340/and64/generic/trusty/release
12M  ./snapshot-20150414-223340/and64/generic/trusty
12M  ./snapshot-20150414-223340/and64/generic
12M  ./snapshot-20150414-223340/and64
1.8G  ./snapshot-20150414-224942/and64/generic/trusty/release
1.8G  ./snapshot-20150414-224942/and64/generic/trusty
1.7G  ./snapshot-20150414-224942/and64/generic/precise/release
1.7G  ./snapshot-20150414-224942/and64/generic/precise
3.4G  ./snapshot-20150414-224942/and64/generic
3.4G  ./snapshot-20150414-224942/and64
8.0K  ./snapshot-20150414-224942/grub
1.8G  ./snapshot-20150414-224942/1386/generic/trusty/release
1.8G  ./snapshot-20150414-224942/1386/generic/trusty
1.7G  ./snapshot-20150414-224942/1386/generic/precise/release
1.7G  ./snapshot-20150414-224942/1386/generic/precise
3.4G  ./snapshot-20150414-224942/1386/generic
3.4G  ./snapshot-20150414-224942/1386
6.0G  ./snapshot-20150414-224942
4.0K  ./snapshot-20150414-224258/and64/generic/trusty/release
8.0K  ./snapshot-20150414-224258/and64/generic/trusty
12K   ./snapshot-20150414-224258/and64/generic
16K   ./snapshot-20150414-224258/and64
20K   ./snapshot-20150414-224258
4.0K  ./snapshot-20150414-223509/and64/generic/trusty/release
8.0K  ./snapshot-20150414-223509/and64/generic/trusty
12K   ./snapshot-20150414-223509/and64/generic
16K   ./snapshot-20150414-223509/and64
20K   ./snapshot-20150414-223509
4.0K  ./cache
4.0K  ./snapshot-20150414-224046/and64/generic/trusty/release
8.0K  ./snapshot-20150414-224046/and64/generic/trusty
12K   ./snapshot-20150414-224046/and64/generic
16K   ./snapshot-20150414-224046/and64
20K   ./snapshot-20150414-224046
6.8G  ./snapshot-20150414-224046
pushpinder@ubuntu: /var/lib/maas/boot-resources$
```

Once the images are downloaded you can now add nodes and PXE boot them using these images. The settings for which image to be used can be configured easily in the user interface. Also, custom boot images can be imported by adding the required sources in the boot-resources-config file. It may be under a different name depending on the Ubuntu build.

These changes should now be visible in the browser user interface:

With Apple pushing out x Accepted clusters | ubuntu x MAAS: Metal As A Service x Interested in MAAS and J x Setting a VMware Workst x Adding nodes to the syst x

192.168.58.107/MAAS/clusters/ root

Nodes Clusters Zones Networks

1 accepted cluster in ubuntu MAAS

1 accepted cluster / 0 pending clusters / 0 rejected clusters

| Name | Managed interfaces | Boot images | Nodes |
|----------------|--------------------|-------------|-------|
| Cluster master | 0 | 12 | 0 |

Import boot images

© 2012 Canonical Ltd. Ubuntu and Canonical are registered trademarks of Canonical Ltd. View Documentation

ubuntu

18:10 14-04-2015

With Apple pushing out x Boot Image list | ubuntu x MAAS: Metal As A Service x Interested in MAAS and J x Setting a VMware Workst x Adding nodes to the syst x

192.168.58.107/MAAS/clusters/a261b08d-c861-424e-97ef-e782ac9e57de/bootimages/ root

ubuntu MAAS

Nodes Clusters Zones Networks

Search nodes

12 boot images in cluster Cluster master

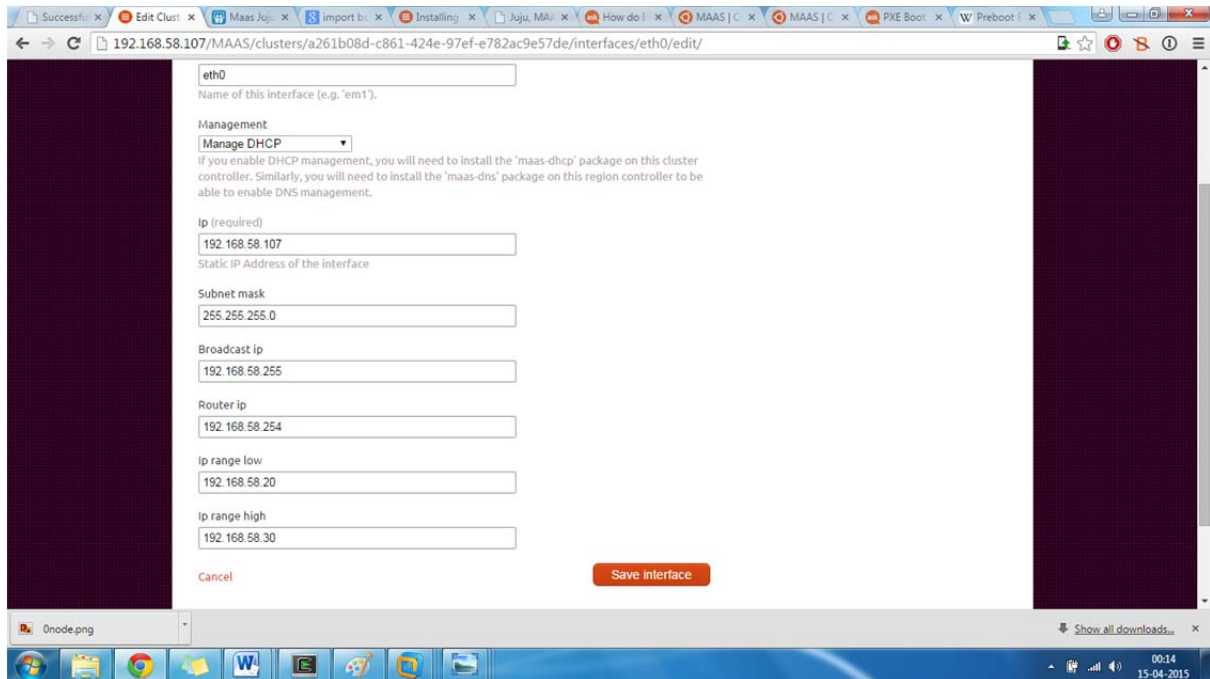
Boot images in cluster Cluster master

| ID | Release | Architecture | Subarchitecture | Purpose | Label | Imported on |
|----|---------|--------------|-----------------|---------------|---------|----------------------------|
| 4 | trusty | amd64 | generic | commissioning | release | April 14, 2015, 11:38 p.m. |
| 7 | trusty | amd64 | generic | install | release | April 14, 2015, 11:38 p.m. |
| 3 | trusty | amd64 | generic | xinstall | release | April 14, 2015, 11:38 p.m. |
| 5 | trusty | i386 | generic | commissioning | release | April 14, 2015, 11:38 p.m. |
| 12 | trusty | i386 | generic | install | release | April 14, 2015, 11:38 p.m. |
| 9 | trusty | i386 | generic | xinstall | release | April 14, 2015, 11:38 p.m. |
| 6 | precise | amd64 | generic | commissioning | release | April 14, 2015, 11:38 p.m. |
| 11 | precise | amd64 | generic | install | release | April 14, 2015, 11:38 p.m. |
| 10 | precise | amd64 | generic | xinstall | release | April 14, 2015, 11:38 p.m. |
| 2 | precise | i386 | generic | commissioning | release | April 14, 2015, 11:38 p.m. |
| 8 | precise | i386 | generic | install | release | April 14, 2015, 11:38 p.m. |
| 1 | precise | i386 | generic | xinstall | release | April 14, 2015, 11:38 p.m. |

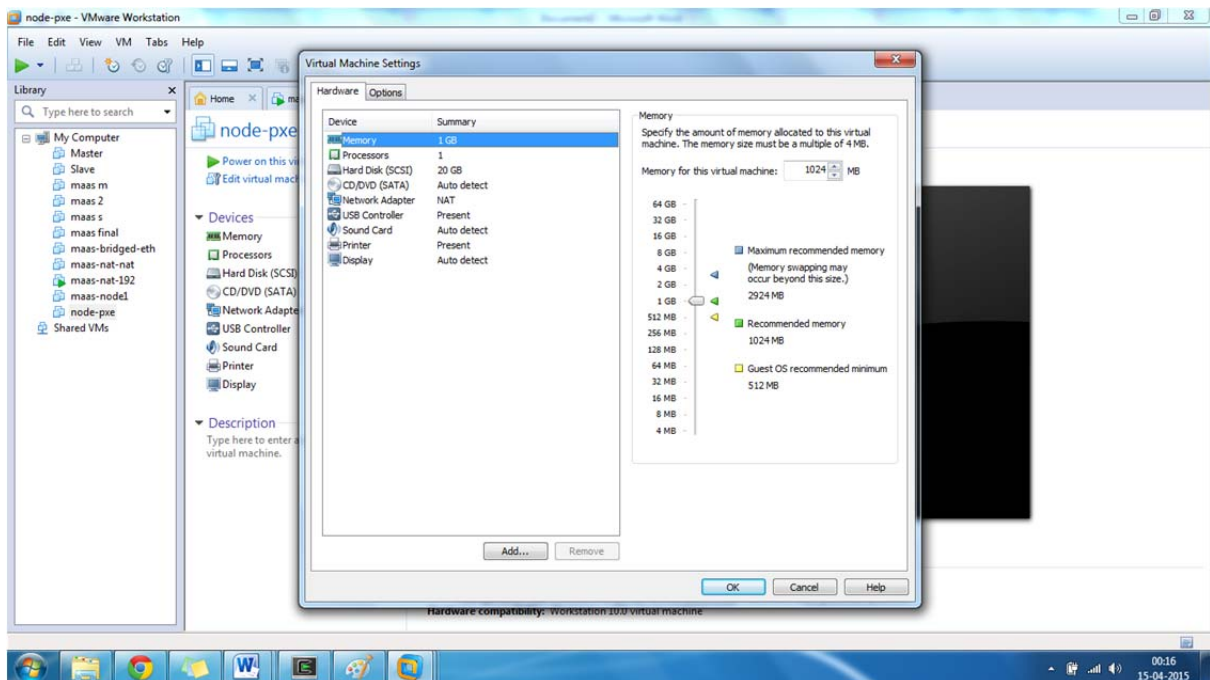
18:11 14-04-2015

Adding nodes:

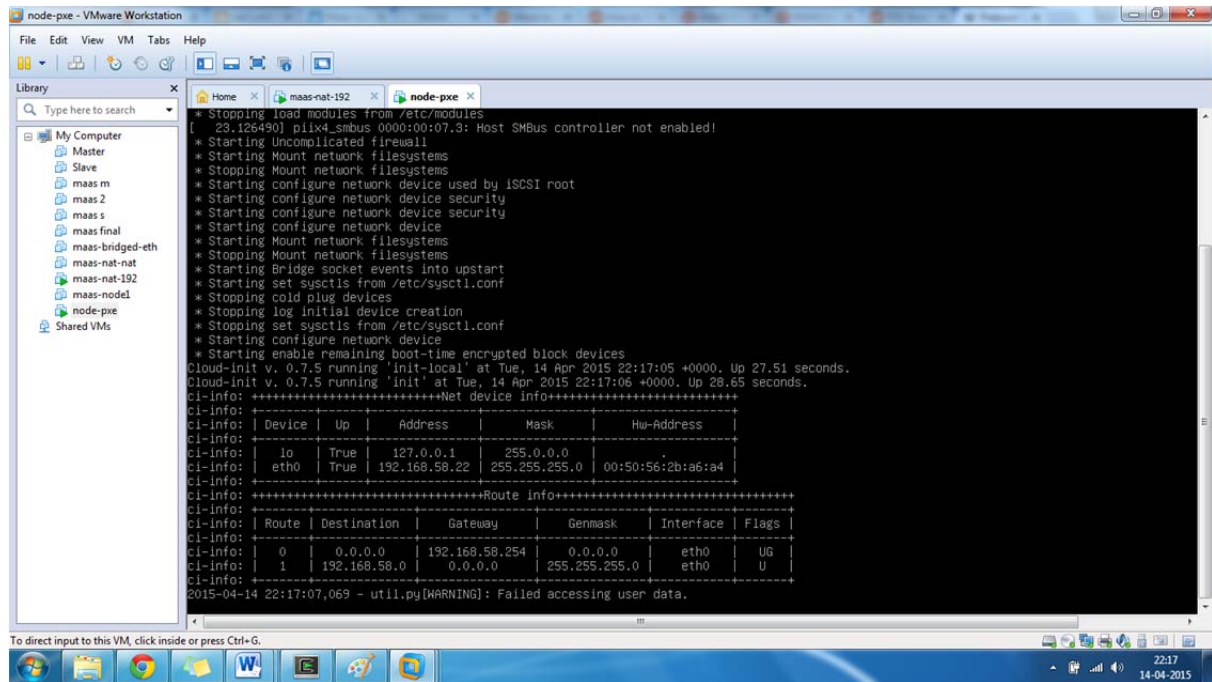
Till now the system has just one cluster, one cluster controller and no nodes. MAAS allows us the flexibility to add nodes using PXE boot over a network connection. For this to happen, the bare metal machine should have PXE boot enable and should be connected to the same network as the maas cluster controller. Also, the maas DHCP server must be enable in order to add nodes. This can be done using the following setting in the user interface:



This DHCP server will assign an IP to any machine which connects to its network. To simulate a bare metal machine, a new VM was created with the following setting:



No boot media is selected so this VM automatically goes for PXE boot over the network on powering on and if the maas DHCP is configured correctly then the VM might ask for a username and password. The VM will show the hostname as maas-enlist. Try username root and password as the one you set up during user creation in the maas cluster controller. It might not directly login and might give errors. The VM will automatically shut down. After multiple tries, the VM showed:

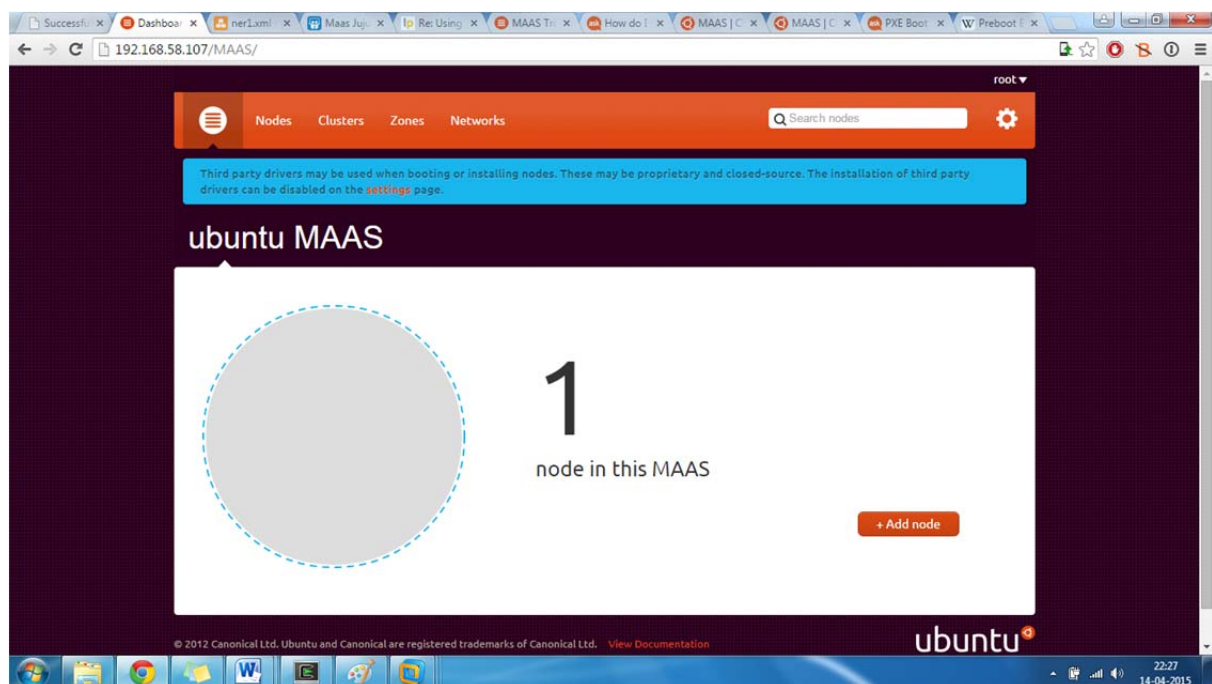


```
node-pxe - VMware Workstation
File Edit View VM Tabs Help

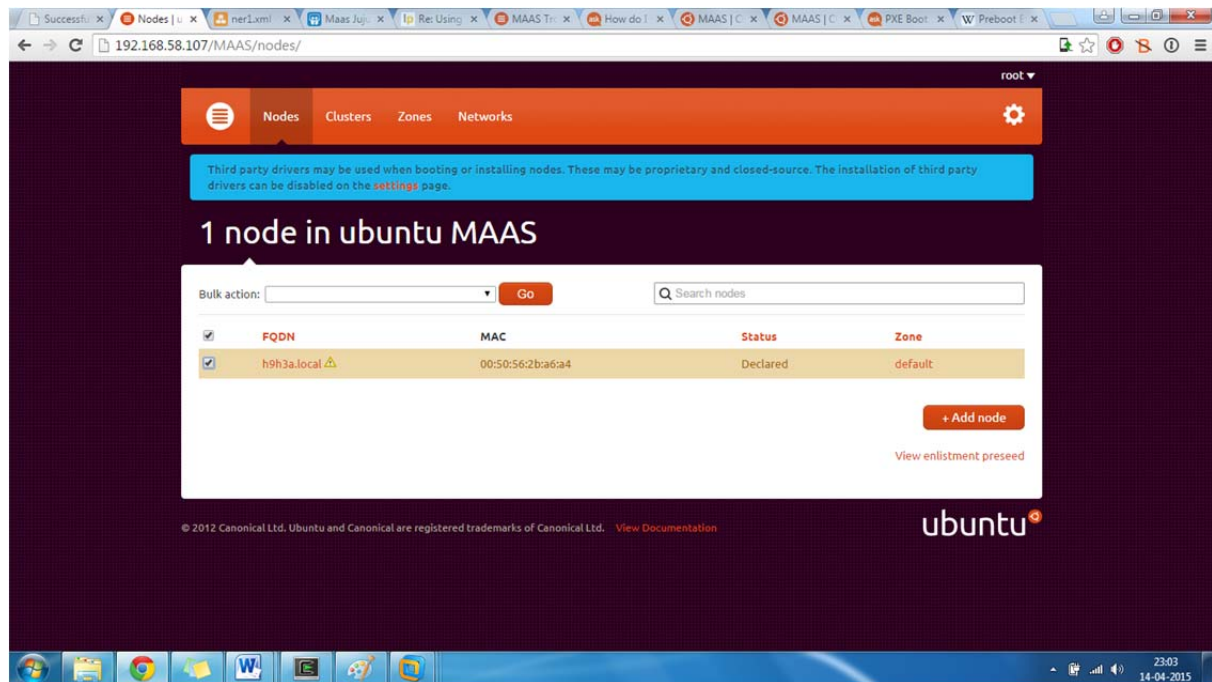
Library
Type here to search
My Computer
  Master
  Slave
  maas m
  maas 2
  maas s
  maas final
  maas-bridged-eth
  maas-nat
  maas-nat-192
  maas-nodel
  node-pxe
  Shared VMs

* Stopping load modules from /etc/modules
[ 23.126490] pll4_smbus 0000:00:07:3: Host SMBus controller not enabled!
* Starting Uncomplicated firewall
* Starting Mount network filesystems
* Stopping Mount network filesystems
* Starting configure network device used by iSCSI root
* Starting configure network device security
* Starting configure network device
* Starting configure network device
* Starting Mount network filesystems
* Stopping Mount network filesystems
* Starting Bridge socket events into upstart
* Starting set sysctls from /etc/sysctl.conf
* Stopping cold plug devices
* Stopping log initial device creation
* Stopping set sysctls from /etc/sysctl.conf
* Starting configure network device
* Starting enable remaining boot-time encrypted block devices
Cloud-init v. 0.7.5 running 'init-local' at Tue, 14 Apr 2015 22:17:05 +0000. Up 27.51 seconds.
Cloud-init v. 0.7.5 running 'init' at Tue, 14 Apr 2015 22:17:06 +0000. Up 28.65 seconds.
ci-info: *****Net device info*****
ci-info: | Device | Up | Address | Mask | Hw-Address |
ci-info: |-----|---|-----|-----|-----|
ci-info: | lo      | True | 127.0.0.1 | 255.0.0.0 | 00:00:00:00:00:00 |
ci-info: | eth0    | True | 192.168.58.22 | 255.255.255.0 | 00:50:56:2b:a6:a4 |
ci-info: *****Route info*****
ci-info: | Route | Destination | Gateway | Genmask | Interface | Flags |
ci-info: |-----|-----|-----|-----|-----|-----|
ci-info: | 0      | 0.0.0.0     | 192.168.58.254 | 0.0.0.0 | eth0      | UG    |
ci-info: | 1      | 192.168.58.0 | 0.0.0.0     | 255.255.255.0 | eth0      | U     |
ci-info: *****
2015-04-14 22:17:07,069 - util.py[WARNING]: Failed accessing user data.
```

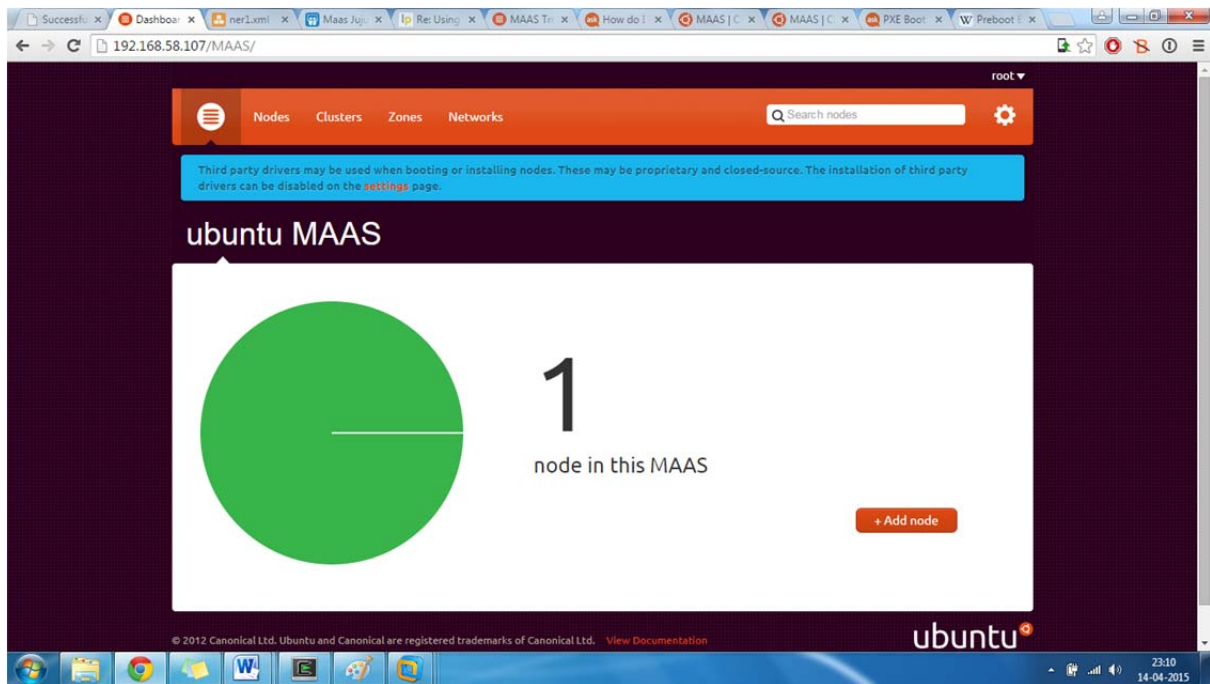
After this, the view in the user interface changes to:



Now the node has been added. The node VM will automatically shut down and the user interface might again show 0 nodes. Restart the node VM and it will now show up again permanently. Now the node has been added to the cluster but has not been commissioned yet. This can be done by clicking on the “Nodes” link on the top. You will get to a screen:



Select the node and choose “commission nodes” from the Bulk Action drop down list. Restart the node VM and it will download some packages from the internet. It might take some time multiple tries, on successful completion, the user interface will show:



Similarly, multiple nodes can be added to the cluster using VM's with PXE boot. In a larger organization, there might be multiple cluster spread over a vast geographic are. In such a case, there can be multiple region controllers and the clusters can be grouped together into zones. Also these clusters can be a part of different networks and the cluster controllers will handle everything seamlessly. All these settings can be configured easily in the user interface. Now the maas cluster is ready and an orchestration layer such as Juju can be added on top to use the nodes productively.

Conclusion:

For harnessing the full power of maas, physical machines should be used but VM's can be a reasonable substitute to show the working of a maas cluster. The installation is quite straight forward but there can a lot of unexpected errors and the documentation is not generally geared for VM deployment. With enough trial and error I was able to get a maas cluster running and enlisted a live node into the cluster. On demand services cannot be done with VM's as they lack Wake On LAN capability so the VM's have to be turned on manually. This report should prove to be a useful resource to anyone trying to deploy a maas cluster in a virtual environment.