Install Ubuntu 12.04LTS on a node, include OpenSSH and Virtual Machine tools

Install python-vm-builder

Become root (sudo -H bash)

```
apt-get update && apt-get dist-upgrade -y && apt-get install python-vm-builder -y
```

Create a bridge interface on your local machine (for some reason using virbr0 doesn't seem to make things happy, probably because that's expected to be used for KVM NAT):

```
cat >> /etc/network/interfaces<<EOF

auto virbr0
iface virbr0 inet static
   address 0.0.0.0
   bridge_ports eth1
   bridge_fd 9
   bridge_stp off
EOF
```

Restart networking:

```
/etc/init.d/networking restart
```

create a vmbuilder default file:

```
cat > /etc/vmbuilder.cfg <<EOF
[DEFAULT]
# Default is 1G tmpfs; Uncomment this line if you've >=2G of free RAM.
tmpfs = suid,dev,size=2G

arch = amd64
#arch = i386
domain = lab
part = vmbuilder.partition
user = localadmin
name = localadmin
pass = ubuntu

[kvm]
libvirt = qemu:///system
#network = br0
bridge = virbr0
virtio_net = true

[ubuntu]
# If you are in the SDU DMZ, you can use my ubuntu mirror:
# mirror = http://192.168.90.10/ubuntu
# If using package cache software (apt-proxy), uncomment line below and set correct IP and Port:
# install-mirror = http://127.0.0.1:9999/ubuntu
# OR better 'apt-get install apt-cacher-ng' and then:
# install-mirror = http://127.0.0.1:3142/ubuntu

suite = precise
flavour = virtual
#components = main,universe,restricted,multiverse
components = main,universe
```

```
# Example adding PPA and installing extra software packages after base OS installation:
#ppa = bcfg2/lucidtesting
addpkg = openssh-server, unattended-upgrades, bcfg2, acpid, ntp, puppet, ipmitool, git
EOF
```

Create a partition model:

```
cat >/etc/vmbuilder.partition<<EOF
root 8192
swap 256
EOF
```

Now build a virtual machine

```
vmbuilder kvm ubuntu --hostname=build-vm --destdir=/build-vm
```

Once built, you can either run the vm from the cli:

```
apt-get install --no-install-recommends ubuntu-desktop -y
apt-get install gnome-session-fallback -y
apt-get install vnc4server -y

vncserver
vncserver -kill :1

cat > ~/.vnc/xstartup<<EOF
#!/bin/sh
# Uncomment the following two lines for normal desktop:
unset SESSION_MANAGER
#exec sh /etc/X11/xinit/xinitrc
gnome-session --session=gnome-classic &
[ -x /etc/vnc/xstartup ] && exec /etc/vnc/xstartup
[ -r $HOME/.Xresources ] && xrdb $HOME/.Xresources
xsetroot -solid grey
vncconfig -iconic &
#x-terminal-emulator -geometry 80x24+10+10 -ls -title "$VNCDESKTOP Desktop" &
#x-window-manager &
EOF
```

Now restart vnc server

```
vncserver -geometry 1280x1024
```

And then install virt-manager (for graphical VM management)

```
apt-get install virt-manager -y
```