

Liquid Galaxy on Virtual Machines

Estimated Time: 1-3 hours

Getting multiple computers can be hard, so sometimes it is preferred to use virtual machine software to emulate having separate computers for Liquid Galaxy. For this tutorial, I used VirtualBox because it was free, open-source, and had good tutorials online, but a similar process should work with other software. Use a computer with at least 60 GB of space on the hard disk and 8 GB of RAM (preferably more).

Step 1: Initial Installation

Download and install [VirtualBox](#) and save the [18.04 Bionic Beaver Linux desktop image](#) to your computer as an .iso file. Open Preferences. In Network, click the green button to create a new NAT Network. Click Edit and give it a Network CIDR of 10.42.42.0/24.

Launch Virtualbox. Click the New button to start configuring the machine.

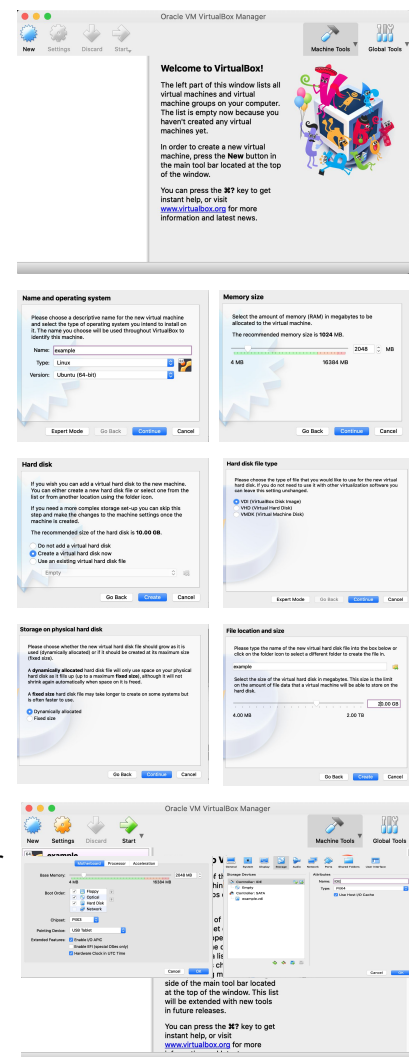
Set Name to something descriptive, like left or lg2. Set Type to Linux. Set Version to Ubuntu (64-bit).

Set Memory Size to 2048 MB. Higher sizes are possible, but setting it lower may result in the virtual machine crashing.

Select the option to Create a virtual hard disk now. Use any Hard Disk file type. If you have a large hard drive, select Fixed size, but if you want to conserve space, use Dynamically allocated instead. Set the Disk size to 15-20 GB, as Liquid Galaxy combined with Ubuntu, at its most minimal installation, will require slightly over 10 GB. When factoring for other programs, it is a good idea to retain extra space, especially on the master machine. This also leaves room to get creative in the future.

Click the Settings button. In System, check Enable EFI. In Storage, click on the disk icon with Empty next to it. Then, click on the disk icon next to the Optical Drive dropdown and select Choose Virtual Optical Disk file. In the file selection dialog, find the saved .iso and select it. In Network, change Attached to to NAT Network with Name as the name of the previously created NAT Network. This gives all machines under the network different IPs so they can communicate.

Save the settings and click on the virtual machine in the menu to power it on.



Step 2: Setting up Ubuntu

This should be the most straightforward part. If a dialog comes up for selecting the disk, choose the 18.04 .iso again.

When the command line menu comes up, select Install Ubuntu.

Continue through the installation process. Check that the keyboard works. Select a normal installation. If you want, you can select to install third party software, although this will take up more space. When given the option, choose to erase the disk and install Ubuntu. This is completely harmless and will not affect the host machine. Click Continue on the dialog for reformatting. Select a time zone.

Name the machine after its position in the frame order, like lg2. Also set this for Computer Name. For Username, set it to lg. Choose a password and continue.

Wait for Ubuntu to install. While you wait, try setting up another virtual machine if you have enough RAM.

Step 3: Installing Dependencies

Launch the Terminal application.

Run the following commands, and enter your password/press Y when prompted.

```
sudo apt install lsb-core
```

```
sudo apt install lsb
```

These packages install functions needed by Liquid Galaxy.

If not already installed, install lightdm with

```
sudo apt install lightdm
```

When the dialog comes up, change the default display manager to lightdm.

If not already installed, install ssh and net-tools with

```
sudo apt install ssh
```

```
sudo apt install net-tools
```

This lets machines communicate with each other and helps check IP addresses.

Install curl with

```
sudo apt install curl
```

This will allow communication and retrieval of information from the internet.

Run the installation command from the [Liquid Galaxy Github repository](https://raw.githubusercontent.com/LiquidGalaxyLAB/liquid-galaxy/master/install.sh).

```
bash <(curl -s
```

```
https://raw.githubusercontent.com/LiquidGalaxyLAB/liquid-galaxy/master/install.sh)
```

Step 4: Installing Liquid Galaxy

The steps for this will differ slightly based on whether the machine is running as the master (lg1) or a slave (the other machines). Complete steps 1-4 for the master first because the slaves need to connect to master for the config files.

Master Installation

Fill in the following values:

Machine id (i.e. 1 for lg1) (1 == master): 1

Total machines count (i.e. 3): how many machines you plan to set up

Unique number that identifies your Galaxy (octet) (i.e. 42): any number is fine

Do you want to install extra drivers? (y/n): y

```
Machine id (i.e. 1 for lg1) (1 == master): 1
usage: grep [-abcDEFGHhIiJLlmnOoqRSsUVvwXZ] [-A num] [-B num] [-C[num]]
          [-e pattern] [-f file] [--binary-files=value] [--color=when]
          [--context[=num]] [--directories=action] [--label] [--line-buffered]
          [--null] [pattern] [file ...]
Save the IP ADDRESS of this machine:
Press any key to continue
Total machines count (i.e. 3): 3
Unique number that identifies your Galaxy (octet) (i.e. 42): 42
Do you want to install extra drivers? (y/n): y
```

Slave Installation

Machine id (i.e. 1 for lg1) (1 == master): the planned position in the frame order

Master machine IP (i.e. 192.168.1.42): run `ifconfig -a` on the master (`sudo apt-install net-tools` may be needed) to get the IP

Master local user password (i.e. lg password): master machine password

Total machines count (i.e. 3): same number as before

Unique number that identifies your Galaxy (octet) (i.e. 42): same number as before

Do you want to install extra drivers? (y/n): y

```
Machine id (i.e. 1 for lg1) (1 == master): 2
Make sure Master machine (lg1) is connected to the network before proceeding!
Master machine IP (i.e. 192.168.1.42): 10.42.42.4
Master local user password (i.e. lg password):
Total machines count (i.e. 3): 3
Unique number that identifies your Galaxy (octet) (i.e. 42): 42
Do you want to install extra drivers? (y/n): y
```

Step 5: Setting up Liquid Galaxy

Wait for all installations to finish. If the installations freeze, restarts may be required. Once installations for both master and slaves are complete, enter the root directory in master using `cd ..` and execute `sudo nano etc/hosts` to edit the hosts file. Change the ip address for lg1 to the master machine's IP address (which should have been found during slave installation). Run `ifconfig -a` on the slaves to find out their IP addresses, and change master accordingly if needed. The IPs should be in the order they were created (ie, the first virtual machine created is 10.42.42.4, the second 10.42.42.5, and so on).

Step 6: Testing the connection

To test whether or not the machines can connect, try running `ssh lg@lg2` and enter the password when prompted. If it works, you should be able to remotely run commands on lg2 from the master machine. Run the command `exit` to log out of the ssh connection. Do the same for the other machines. Next, navigate to `/earth/scripts` on each machine and run `write-drivers-ini.sh`. You should get an output like this, with the yaw value differing for different frames and master being true on the master machine.

```
MASTER: false
SLAVE: true
VSYNCHOST: 10.42.42.255
VSYNCPORT: 45678
YAW: -42
FOV: 36.5
NAV: /dev/input/spacenavigator
QUERY: /tmp/query.txt
```

Then launch Google Earth on each of the machines. If the globe starts spinning with two arrows overlaid on top, go to `Tools>Options>Navigation` and uncheck `Enable controller`. This changes the input device from a spacenavigator to a mouse. To change this permanently, go into `drivers.ini` in `earth/builds/latest` and comment out the lines that start with `SpaceNavigator` by putting a `;` (semicolon) before them.

If the displays are slightly off, this may be due to the yaw value. This can also be fixed by changing the `ViewSync/yawOffset` value in `drivers.ini`. For a three-machine setup, this is achieved by changing 42 and -42 to 36.5 and -36.5, but I don't know the values for other setups.

Once you are finished editing `drivers.ini`, relaunch Google Earth.

With a lot of luck, you should get something like this:

