PC and Virtual Box Liquid Galaxy installation

Made with the help of GCI 2019-2020 students:
Emilie (Kewbish), Dylan, Merul Dhiman, Asmrvin, Alvii_07
Thank you !!

Covers both installations, using regular pcs and using vms with Virtual Box.
Other technologies can be used, as bluestacks for emulating and Android tablet.

Note: this document has NO warranties after all, there're plenty of methods to do this, and also there can be errors due to every installation different system or networks configuration.

Mr. Worldwide, our new mascot
Created by student Kahi
2D version by student Emilie

Our main web site:

Email: liquidgalaxylab@gmail.com

Introduction to Liquid Galaxy

The Liquid Galaxy system is an awesome **panoramic display** of a cluster of computers, usually three or more. Liquid Galaxies have been installed around the world, particularly in use at museums or universities. The Liquid Galaxy system is based off Google Earth, and started as an open source Google 20% project. Now, it's working on multiple projects all revolving around the Liquid Galaxy system, such as LGxEDU and WikiMedia. Use cases for the Liquid Galaxy include education about geography, visualizing tours and points of interest around the world as a virtual tourist, displaying information about cities, and exploring the globe in general! For example, museums can use the Liquid Galaxy to visualize data about certain cities and placemarks around the world, and teachers can bring their students right into a country's history from their classroom.

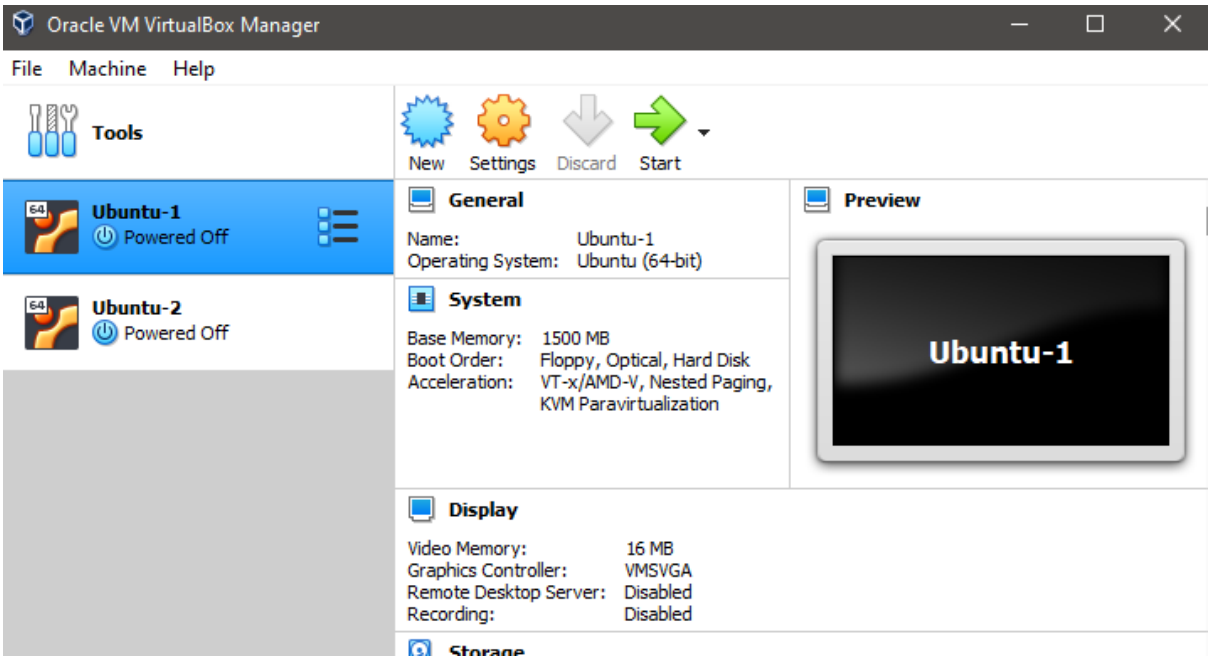
Interested in building a Liquid Galaxy for yourself? It can be done with a single computer, using a technology called **virtual machines** - when a computer emulates another computer.

Installing VirtualBox & Prepackaged ISOs

VirtualBox is a Virtual Machine (VM) software that completely free and open source. From their website, choose the latest install version for your host OS. (Host refers to the computer you install VirtualBox on, slaves refer to the computers created via VirtualBox). Follow the installation instructions (default installation settings are fine!). When VirtualBox loads up, proceed to the third paragraph below.

You may need to enable Virtualization Technology on your laptop. On some systems, this resides in the laptop settings or manufacturer settings, but you will need to look up how to do this for your system.

Once VirtualBox is up and running, you should see a screen like this:



(Without the two VM's on the side, but we'll be adding to your VM list soon!)

Next, click the New button, which looks like a blue badge. When the wizard opens up, change the name and location to whatever you want, but choose Linux as the type and Ubuntu as the version. Allocate 1500MB - 2048MB of RAM (ensure your host can support all three VMs with 2GB for host system and graphics to spare) to the VM. If you'd like to manually install Liquid Galaxy, create a new hard drive in the next step, and proceed to the Manual Installation section. If you'd like to easily install Liquid Galaxy, proceed to the Automatic Installation section.

Automatic (Snapshot) Installation

However, if you'd just like to try the system out, select 'Use an existing hard drive'.

If you'd like to use our prepackaged isos, download and unzip the files from this Drive link. When you reach the hard drive step of a new VM install, navigate to and select Ubuntu-1's iso. Continue with installation with default settings.

Repeat these instructions, adding a different iso e-ach time. In the end, you should have three VMs, one for each of the isos included in the Drive file.

Try booting each VM. If it asks for an ISO to boot from, re-select the iso you used for the install. For example, if it asks for an iso to boot from, and you're on the VM that you selected Ubuntu-2 for, re-select the Ubuntu-2 iso.

If everything's gone well, you should be able to boot all three and open Google Earth on each. If they don't sync up, check the installation tips below. Particularly, check the drivers.ini of each, ensuring that the IPs match up and the VMs can ping to each other. Ensure the ports are the same on each drivers.ini, and make sure that the address matches the broadcast address you receive when you run `ifconfig` in Terminal. Look for the numbers after Bcast or Broadcast, which should end in .255. If other errors pop up, refer to the community tips below.

Manual Installation

Select **VDI** for the file type and **Dynamically Allocated** for the Storage on Physical Hard Disk Option. These choices will make the virtual hard drive change size as it fills up, as opposed to it taking up space while being partially empty. Next, keep the default location and the default size, and click create.

Congrats! You've made your first VM!

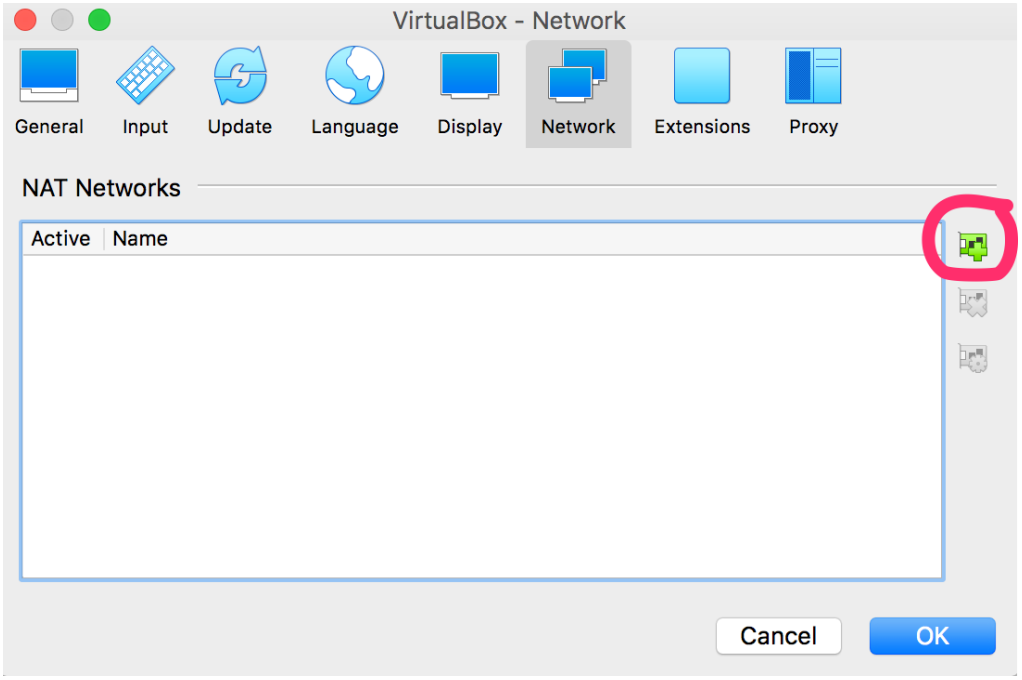
Now, let's make two more (because Liquid Galaxy needs at least three to be *really* cool). Once again, click new and follow the earlier instructions, only changing the name. Next, repeat that one last time, ending with a total of three VMs in your sidebar.

Now, continue to the Setting Up the Network section.

Setting Up the Network

We're next going to be creating a Network Address Translation Network (NAT Network), so that all of the VMs can talk to each other. In VirtualBox, open preferences and navigate to the network tab.

Now, select the button in the picture below to create a new network.



After pressing that button, select the button two below that looks similar but with a gear icon. Name the network whatever you want, and make sure to check Supports IPv6 and Supports DHCP. Now, press OK to leave the popup and OK to leave the preferences page. Now, for each VM, select it on the left, click the settings button, go into network, and change “Attached To:” to **Nat Network**. Your network’s name should pop up in the “Name:” field. Then, once you go back into the network settings of your VM, click on network settings and select “NAT network”, then select your NAT network that you just made. Depending on whether you would like a drivers.ini (easier setup, but cannot work with the LG apps) or an install.sh (a little more difficult, can work with LG apps) setup, follow the appropriate steps.

Drivers.ini

Once your three VMs have been installed, you can install Google Earth and create a drivers.ini setup.
Pros: easy and intuitive to set up
Cons: will not work with Liquid Galaxy apps

For this installation, we’ll assume the **master** is the main VM in the middle of the setup, and the **slaves** are the VMs on either side.

Alternatively, install Google Earth from the Google website. When you’ve downloaded the .deb file, you can install it by double-clicking on the file in File Explorer. This will open up a Software Center tab, where the installation will finish.

Repeat the Google Earth install on all three VMs. Ensure Google Earth works on all three VMs before proceeding.

Next, on the master, run ‘sudo apt install net-tools’ and ‘ifconfig’ to get the broadcast address. This should be after the Bcast or broadcast options and will end in .255. Write the IP down somewhere, as we’ll be needing it in the next steps.

On the master VM, navigate to opt/google/earth/pro in Terminal (use ‘cd /opt/google/earth/pro’) and run ‘sudo nano drivers.ini’ to edit the file. Between the first { and the first }, insert the following lines:

```
ViewSync/send = true;
ViewSync/receive = false;

ViewSync/hostname = BROADCAST_ADDRESS
ViewSync/port = 21567

ViewSync/yawOffset = 0;
ViewSync/rollOffset = 0;
ViewSync/pitchOffset = 0;
ViewSync/horizFov = 36.5;
```

On the first slave VM, navigate to the same directory and add the same lines to the drivers.ini. However, remove the ViewSync/hostname = BROADCAST_ADDRESS line, and change the yawOffset line to ViewSync/yawOffset = 36.5;

On the second slave VM, navigate to the same directory and add the same lines to the drivers.ini. As with the other slave, remove the hostname line, but change the yawOffset line to ViewSync/yawOffset = -36.5;

Run Google Earth on all three VMs. If everything’s gone well, the slaves will sync to either side of the master VM.

Install.sh Setup

Once your three VMs have been installed (fresh VMs, don’t use install.sh after installing Liquid Galaxy with the drivers.ini method, or vice-versa), you can use the shell script (series of commands that we’ve premade to make installation easier) to install Liquid Galaxy.
Pros: works with other LG media, such as the apps
Cons: can be a little tricky, tested on Ubuntu only

For this installation, we’ll assume the **master** is the main VM in the middle of the setup, and the **slaves** are the VMs on either side. As well, we’re assuming that all three VMs are running Ubuntu **16.04**.

Ensure all three VMs are up to date by running ‘sudo apt update’ and ‘sudo apt upgrade -f’ to update all libraries.

On all VMs, open Terminal (click Unity icon, search and click on Terminal icon). Run ‘sudo apt install curl’ to install curl, a program to get content from the internet. Once this has been done, run ‘sudo apt install lsb lsb-core’ to install libraries that Google Earth requires to run. Accept installation with ‘Y’ when prompted. Repeat for each of the three VMs.

On the master VM, run this command:
‘bash <(curl -s https://raw.githubusercontent.com/LiquidGalaxyLAB/liquid-galaxy/master/install.sh)’
This will get an installation script from the Liquid Galaxy GitHub. It is safe to use, and is open source, if you’d like to verify there is no malware.

During master installation, several prompts will come up. Enter 1 when prompted for master, 3 when prompted for amount of machines, 42 when prompted for an octet, and ‘n’ for extra drivers. Enter ‘y’ if prompted if you want to continue. You will also be prompted for SSH keys and passphrases. Press Enter twice, without entering any alphanumeric information.

Ensure Liquid Galaxy has been installed on master, then proceed to slave installation.

On the master VM, run ‘ifconfig’ in Terminal to get the machine IP. This will follow the inet addr section, and will look like x.x.x.x (where x are numbers).

On the first slave VM, open Terminal and run:
‘bash <(curl -s https://raw.githubusercontent.com/LiquidGalaxyLAB/liquid-galaxy/master/install.sh)’
Fill in 2 when prompted for machine ID, the master machine IP when prompted for the master IP, the master password when prompted for the password, 3 when prompted for total machines, and 42 when prompted for octet number. If an extra drivers question comes up, enter ‘n’. Enter ‘y’ if prompted if you want to continue during the installation process. You may also be prompted for SSH keys and passphrases. Press Enter twice, without entering any alphanumeric information.

On the second slave VM, repeat the steps for the first slave VM. However, change machine ID to 3. The rest of the steps will remain the same.

Once all the installations are finished, run ‘lg-relaunch’ in the master’s terminal. If everything has gone well, when you open Google Earth on the three VMs, they should sync up.

Minor bug: if the VMs are too offset, run ‘sudo nano /opt/google/earth/pro/drivers.ini’ to change the ViewSync/yawOffset = 42; line to ViewSync/yawOffset = 36.5; (ensure the signs are the same, i.e. if the yawOffset was -42, the yawOffset should be changed to -36.5). Relaunch Google Earth.

Minor bug: if the LG installation keeps spinning, navigate to Tools > Options > Navigation > Enable Controller and uncheck the checkbox.

LG Installation in 3 different computers

Before Starting:

This is a complete guide for installing Liquid Galaxy in three different computers, not using Virtual Machines, as other guides show. This installation is the Drivers.ini method. You can also install the install.sh one, but here the drivers will be shown.

One of the Liquid Galaxy Google Code-in 2019 students did a quite good guide that you can follow step by step and see all you have to do for installing Liquid Galaxy in that way. We will try to help as we can!!

Installation:

1: Requirements

You should have three different computers. It's not necessary to have very powerful ones, with a normal one, it may work. You should know that this system works with a main PC called master and the other 2 that follow master instructions.

It's VERY important to know that for working Liquid Galaxy system is required an INTERNET CONNECTION, the same in all computers!!

2: Install Google Earth in all computers

You have to enter to Google Earth's page and download the latest version of the program. Install it in the 3 computers. Remember that downloading this program is FREE.

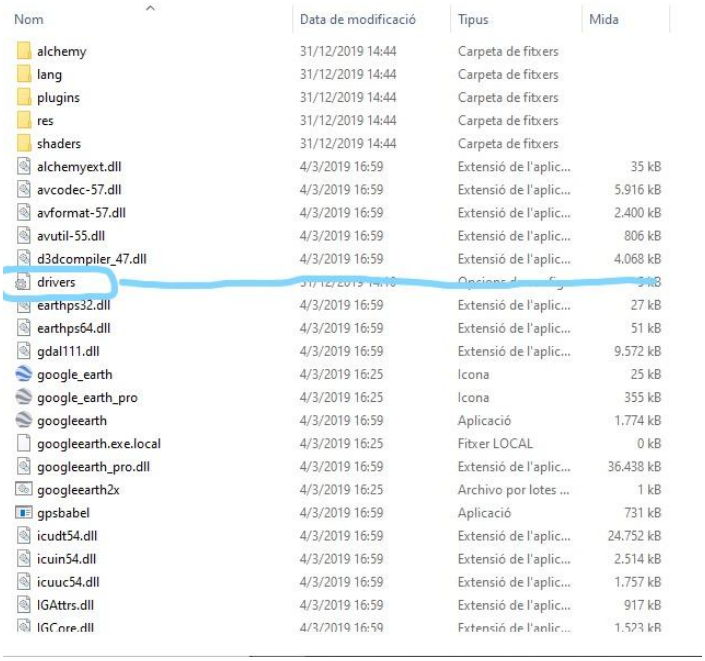
First, it will download an executable file and when that is downloaded, click on it for installing the whole program.

3: Changing Drivers.Ini (Master Computer)

Depending of the OS you have it should change only a bit. In my case, I used Windows, but you can use Mac, Linux…

You should have to enter to the drivers.ini file in the Google Earth program. For doing this you have to enter your Windows Browser and enter your PC files:

PC < Program Files < Google < Google Earth Pro < client < Drivers.ini



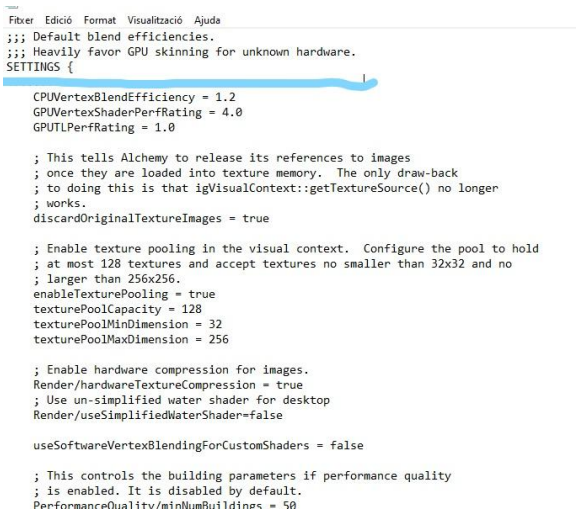
Nom	Data de modificació	Tipus	Mida
alchemy	31/12/2019 14:44	Carpeta de fitxers	
lang	31/12/2019 14:44	Carpeta de fitxers	
plugins	31/12/2019 14:44	Carpeta de fitxers	
res	31/12/2019 14:44	Carpeta de fitxers	
shaders	31/12/2019 14:44	Carpeta de fitxers	
alchemyext.dll	4/3/2019 16:59	Extensió de l'aplic...	35 kB
avcodec-57.dll	4/3/2019 16:59	Extensió de l'aplic...	5.916 kB
avformat-57.dll	4/3/2019 16:59	Extensió de l'aplic...	2.400 kB
avutil-55.dll	4/3/2019 16:59	Extensió de l'aplic...	806 kB
d3dcompiler_47.dll	4/3/2019 16:59	Extensió de l'aplic...	4.068 kB
drivers			
earthps32.dll	4/3/2019 16:59	Extensió de l'aplic...	27 kB
earthps64.dll	4/3/2019 16:59	Extensió de l'aplic...	51 kB
gdal111.dll	4/3/2019 16:59	Extensió de l'aplic...	9.572 kB
google_earth	4/3/2019 16:25	Icona	25 kB
google_earth_pro	4/3/2019 16:25	Icona	355 kB
googleearth	4/3/2019 16:59	Aplicació	1.774 kB
googleearth.exe.local	4/3/2019 16:25	Fitxer LOCAL	0 kB
googleearth_pro.dll	4/3/2019 16:59	Extensió de l'aplic...	36.438 kB
googleearth2x	4/3/2019 16:25	Archivo por lotes ...	1 kB
gpsbabel	4/3/2019 16:59	Aplicació	731 kB
icudt54.dll	4/3/2019 16:59	Extensió de l'aplic...	24.752 kB
icuin54.dll	4/3/2019 16:59	Extensió de l'aplic...	2.514 kB
icuuc54.dll	4/3/2019 16:59	Extensió de l'aplic...	1.757 kB
IGAttrs.dll	4/3/2019 16:59	Extensió de l'aplic...	917 kB
IGCore.dll	4/3/2019 16:59	Extensió de l'aplic...	1.523 kB

We give a picture. Here it says “drivers” because we changed its name. Open the file.

Now, you should have to go to [Liquid Galaxy Quickstart webpage] and copy that text:

```
; ViewSync settings
ViewSync/send = true
ViewSync/receive = false

; If send == true, sets the IP where the datagrams are sent
; Can be a broadcast address
ViewSync/hostname = SLAVE_IP_GOES_HERE
ViewSync/port = 21567

; For video caves, we typically want the slave screens to look to the
; left or right (yawOffset) of where the master is looking
ViewSync/yawOffset = 0
ViewSync/pitchOffset = 0.0
ViewSync/rollOffset = 0.0
ViewSync/horizFov = 36.5
```

Paste it in the blue line (there it won’t be blue, we only edited it…):

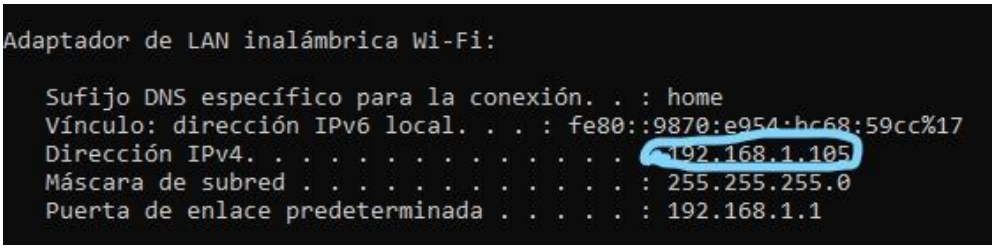


```
Fitxer  Edició  Format  Visualització  Ajuda
;;; Default blend efficiencies.
;;; Heavily favor GPU skinning for unknown hardware.
SETTINGS {

    CPUVertexBlendEfficiency = 1.2
    GPUVertexShaderPerfRating = 4.0
    GPUTLPerfRating = 1.0

    ; This tells Alchemy to release its references to images
    ; once they are loaded into texture memory.  The only draw-back
    ; to doing this is that igVisualContext::getTextureSource() no longer
    ; works.
    discardOriginalTextureImages = true

    ; Enable texture pooling in the visual context.  Configure the pool to hold
    ; at most 128 textures and accept textures no smaller than 32x32 and no
    ; larger than 256x256.
    enableTexturePooling = true
    texturePoolCapacity = 128
    texturePoolMinDimension = 32
    texturePoolMaxDimension = 256

    ; Enable hardware compression for images.
    Render/hardwareTextureCompression = true
    ; Use un-simplified water shader for desktop
    Render/useSimplifiedWaterShader=false

    useSoftwareVertexBlendingForCustomShaders = false

    ; This controls the building parameters if performance quality
    ; is enabled. It is disabled by default.
    PerformanceQuality/minNumBuildings = 50
```

When it's pasted, you will see that there's a line (that you added) that says SLAVE_IP_GOES_HERE. You have to erase it and copy the Broadcast Address.

So for finding the Broadcast Address of your network, you will have to open the CMD program of your computer. Just write "CMD" in your program searcher and the commander will be opened. There, write the command "`ipconfig`". It changes depending of the OS, in windows is "ipconfig", Linux "ifconfig"...try both in your OS.

It will appear many numbers, remember or copy the number *(IPv4 number)* that will appear on this place:



```
Adaptador de LAN inalámbrica Wi-Fi:

   Sufijo DNS específico para la conexión. . : home
   Vínculo: dirección IPv6 local. . . : fe80::9870:e954:bc68:59cc%17
   Dirección IPv4. . . . . . . . . . . . : 192.168.1.105
   Máscara de subred . . . . . . . . . . : 255.255.255.0
   Puerta de enlace predeterminada . . . . . : 192.168.1.1
```

(Will read IPv4 Address on English systems.)
An important thing: That number would be the internet address and you need the broadcast address. For changing it, you only have to change the last number (105 in my case) to 255 (it always finishes by 255). So, my broadcast address would be *192.168.1.255*.

Copy your broadcast address and paste it in the line we said before (Slave_Ip…).

```
;;; Default blend efficiencies.
;;; Heavily favor GPU skinning for unknown hardware.
SETTINGS {
; ViewSync settings
ViewSync/send = true
ViewSync/receive = false

; If send == true, sets the IP where the datagrams are sent
; Can be a broadcast address
ViewSync/hostname = 192.168.1.255
ViewSync/port = 21567

; For video caves, we typically want the slave screens to look to the
; left or right (yawOffset) of where the master is looking
ViewSync/yawOffset = 0
ViewSync/pitchOffset = 0.0
ViewSync/rollOffset = 0.0
ViewSync/horizFov = 36.5
    CPUVertexBlendEfficiency = 1.2
    GPUVertexShaderPerfRating = 4.0
    GPUTLPerfRating = 1.0

    ; This tells Alchemy to release its references to images
    ; once they are loaded into texture memory.  The only draw-back
    ; to doing this is that igVisualContext::getTextureSource() no longer
    ; works.
```

Something similar to this would appear to you, with a different Broadcast Address.

That would be all to do in the master.

4: Changing Drivers.Ini (Slaves Computer)

Copy the same paragraph than before and add it to the slaves Drivers.ini file. You have to enter the drivers.ini file of Google Earth too and paste the paragraphs (same as the master at the moment).
Then, it's also very important to DELETE this line in the drivers file (in both slaves)

```
; If send == true, sets the IP where the data
; Can be a broadcast address
ViewSync/hostname = SLAVE_IP_GOES_HERE
ViewSync/port = 21567
```

After this, you should change this parameters. You have to change true to false and false to true in the send/receive parameters:

```
SETTINGS {
; ViewSync settings
ViewSync/send = true
ViewSync/receive = false
```

Before changing

```
SETTINGS {
; ViewSync settings
ViewSync/send = false
ViewSync/receive = true
```

After Changing parameters

Do the same on both slaves.

Finally, you have to change the *ViewSync/yawOffset* version. It's in number "0". For the synchronization of both slaves to the master, you have to change this option. In the left slave, you should have to change "0" to "36.5" and in the right one change 0 into "-36.5".

72	36.5	0	-36.5	-72

5: Open Google Earth in all computers:

Now, open Google Earth program and see what's happening. Remember to be connected to the same wifi and watch out that your internet is working correctly.

When you open the program in the master, automatically slaves synchronize to the master and the Liquid Galaxy Installation is FINISHED!!

Installing and using Liquid Galaxy Android Apps - LGxEdu, Liquid Galaxy Controller

Using a physical Android tablet

The LGxEDU app is an Android application for Tablets where students can play educational games interacting with the Liquid Galaxy. Furthermore, teachers can create and share new content in this application.

The following guide assumes that the Liquid Galaxy is running and working, and had been installed with the install.sh method.

On your Android tablet, go to the App Store, and search for and download the Liquid Galaxy Controller app. Alternatively, download the LGxEDU app, which should be by the same publisher. Either app will work for this installation.

Open the app. In the upper right corner, click the hamburger menu, and select Administrator Settings. The default password is 'lg'. Again, click the hamburger menu and select Settings. There, you'll find a menu with information about the Liquid Galaxy you'll need to fill in.

Set the IP to the master IP of your Liquid Galaxy. This can be found using `ifconfig` and looking for the eth0 address.
Set the port to the port Liquid Galaxy uses. This can be found by running `sudo nano /opt/google/earth/pro/drivers.ini` and looking for the ViewSync/port line. The port number that follows will be what you enter into the port in the app.
Set the SSH-USER and password to the username and password of the master computer. For example, USER would be lg, and password whatever you set as password.

The default KML API IP should work, but alternatively use the master IP and the port provided.
If using a non-chromebook setup, you'll need to ensure the LiquidGalaxyChromebookInstallation is turned to false.

Now, your Liquid Galaxy should sync when you select any POI on the app, or navigate with touch on the LGxEDU app.

A small known bug has an issue with the controller on the Liquid Galaxy. If the LG is working and syncing but always rotating, navigate to Tools > Options > Enable Controller > False in Google Earth on the master.

Note: the correct port to use for your master Liquid Galaxy VM is 45678, not 21567.
This information can be verified by running `sudo nano /opt/google/earth/pro/drivers.ini` and checking the master IP and ports.

Add-on: Wikimedia Data Project

The Wikimedia Data Project is an additional project for the Liquid Galaxy that provides additional data in KMLs and tours and an online dashboard. It was developed by Guillem Barbosa as a Google Summer of Code project.

Ensure you have an **install.sh** setup, running Ubuntu 16.04. Check the instructions above for more details.

Installing the Project

Open Terminal, and ensure git, Python3, and virtualenv are downloaded. If not, install them by using `sudo apt install git python3 virtualenv`.

Run `git clone https://github.com/gbarbosa4/WikimediaDataProject.git` to download the project files. `cd WikimediaDataProject` to enter the project folder.

Create a virtualenv for fresh Python library installs. This can be done by running `virtualenv -p python wdp`, `sudo apt-get install libffi-dev libssl-dev python3-dev`, and `source wdp/bin/activate`. These will initiate and activate a virtual environment in the wdp folder.

You'll need to edit the requirements.txt file to remove several outdated libraries. These may be removed in a patch in the future. Run `nano requirements.txt` to edit the file. Remove the urllib line, and add a line `whitenoise==3.3.1`.

Install the project requirements by running `pip install -r requirements.txt`.

Run the project by running `./WDLG-Start [Liquid Galaxy IP]` with the Liquid Galaxy IP of the master. For example, if the LG master is at 192.168.0.31, run `./WDLG-Start 192.168.0.31`. Open the web dashboard by opening http://[Your IP]:8000 in your browser. Your IP can be

found through the `ifconfig` command and looking for the eth0 line. It'll look something like 192.168.x.x. Open the IP and port in a web browser.

Known issues: If an issue with the pip installs happens, manually install the correct version of each library, consulting the requirements.txt file. For example, if there's an issue with CairoSVG, manually run `pip install cairosvg==”1.0.19”`.

If an issue with a pykml error appears in the Terminal, you may have to manually add brackets to a library file. Either run `pip install --upgrade pykml` or navigate to the file specified in the error. Find the line with a print statement WITHOUT BRACKETS, and add brackets around the statement so it looks like `print([variable here])` instead of `print variable`

When you open your Liquid Galaxy, the view should now be updated with a new KML.

Liquid Galaxy Installation Tips - by the community!

Drivers.ini Method

1: Use a NAT network for testing, but bridged adapter (in VM settings) if you'd like to use apps (LGxEDU, LGController, OSC controller, etc.)

2: If using Bridged **set promiscuous mode to allow all.**

3: If using apps or other LG media like Wikimedia make sure you're using a install.sh setup and not one with a drivers.ini edit, as the scripts will look for files available in an install.sh setup.

5: If nothing's syncing or the Google Earth appears black try restarting the VMs, sometimes you've started the slave install before the master install or other minor bugs exist, which can usually be fixed with a relaunch.

6: If the slaves don't sync, check the ports in the drivers.ini and check everything can ping via the ping command - ping x.x.x.x -b and ping x.x.x.x to check broadcast and individual addresses.

7: If the slaves aren't connecting, make sure that they are on the same NAT or bridged network, which can be found under preferences of VirtualBox. Ensure that they are connected on the same WiFi network (or Ethernet.)

8: Use the broadcast address in both drivers.ini method and install.sh method.

9: To find the broadcast IP, type in “ifconfig”, then search for the numbers after “broadcast”. It should end with .255

Install.sh Method

1: You may have to manually enable port forwarding in your router settings to forward traffic from port 22 to another port (check router sticker for admin setting website).

2: If the slaves are still not connecting to the master, then disable firewall on master and slaves by running two commands:

sudo ufw disable
sudo iptables -F

3: If the slaves and the master still aren't connecting, then type in the command: sudo gedit /etc/ssh/sshd_config. Check if a "#" is in front of the Port 22 line; if so, delete it.

4: If the slaves are offset a little too much (e.g. 42 degrees instead of 36.5), try manually editing the drivers.ini to adjust the yawOffset to 36.5 or -36.5 (depending on which slave). [An open pull request that will fix this issue is linked here]

General Troubleshooting for Apps

1: Use the install.sh method for testing apps as most LG apps rely on the KML API to move google earth to the desired destination.

2: Ensure your KML API is working properly by visiting http://localhost:PORT/kml/flyto/10/10/1000 (Replace "PORT" with the PORT KML API is installed and configured onto)in a web browser. If the Liquid Galaxy flies to the specified location, everything should be working.

3. The 3 PC's doesn't connect each other so, I think you have installed the Google Earth version 7.3 and you should have 7.1.8 in order that the Liquid Galaxy works.You can prove it by going in Google Earth at Help->About Google Earth. If so, here are the steps to fix it:

1. Install again the Liquid Galaxy
2. When installation is done don't press ENTER button to reboot the VM's
3. Before that, go to Github where liquid galaxy is and download google-earth-pro7_1.deb file
4. Open a new terminal and go where you have downloaded this file and make dpkg -i google-earth-pro7_1.deb . This way you downgrade the Earth version
5. Now you can press ENTER on the other terminal and the Liquid Galaxy will work

4. Show information only on slaves screens

First, you have to go to master at /var/www/html/kml/ and here you have a .kml file for each slave. You only have to change the content in this .kml file with your .kml content.

Next, if you can't see any changes you have to go to the slave and select View -> Sidebar. Here, the .kml file for the slave appears on "Solo KML". You have to select this, right click and go to Properties. Here, replace on "Link", lg1 with the master IP. For example, http://master_ip:81/kml/slave/slave_[slave_number].kml .

If you can't see the changes immediately make sure the in Refresh tab you have selected: Time-Based Refresh and select 'When" : Periodically and select the time you want to refresh the slave kmls.