

CADnetwork Render-Guide

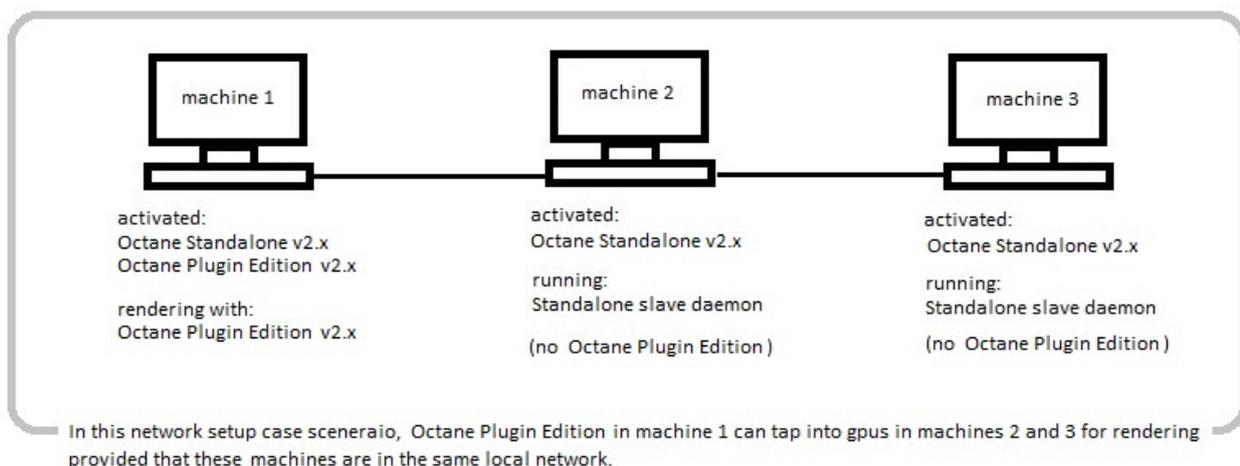
Network Rendering for Octane Render

Network rendering allows additional computers to be utilized in rendering images. OctaneRender distributes compiled render data and not scene data, so no file management is required by the user. Conceptually it is similar to working with additional GPUs by allowing the distributed rendering of single images over multiple computers connected through a fast local area network. Network rendering requires a master and one or more slaves on different computers. The OctaneRender instance that drives the rendering is referred as the “master” and the OctaneRender instances that are helping are referred as the rendering “slaves”.

Since an OctaneRender slave currently requires an activated Standalone license, it is advisable to run the Standalone first to activate a Standalone license on that computer, if necessary. It is best to copy the whole folder of the released archive onto the slave computer. Also ensure that the master and the slave are not blocked by the Operating System firewall or any firewalls in the network. This can be done, for example, by turning off the firewall for home/work networks on the master. If that does not help, also try switching off the firewall on the slave computer for home/work networks.

Master, Slaves and Daemons

The Standalone version or the **octane.exe** act as master and a special console version of OctaneRender, **octane_slave.exe**, can run on other computers as slaves. Of course, they should be all on different computers or they would have to share the same GPUs.



Setting Up The Slave Daemon

To set up the daemon, simply run the batch script `_install_daemon.bat` on the slave computer. During the setup, it will ask which port the daemon should listen to for master requests. After that, the daemon will be resident on that machine and it will be active at all times.

When a daemon is invoked by a master, the slave is quickly launched to get some information about the number of GPUs, version, bitness, etc. and then closed again. After that there is no slave process running. So the daemon just sits there and waits for masters (there could be multiple masters in the local network) to detect it, by scanning the complete local network in regular intervals. The daemon should appear in the daemon list of the network preferences of the masters. If it does not, it can have the following reasons:

The network rendering in the master is not enabled.

The daemon is listening on a different port than the master is scanning. In your case both are 48000, so that's fine.

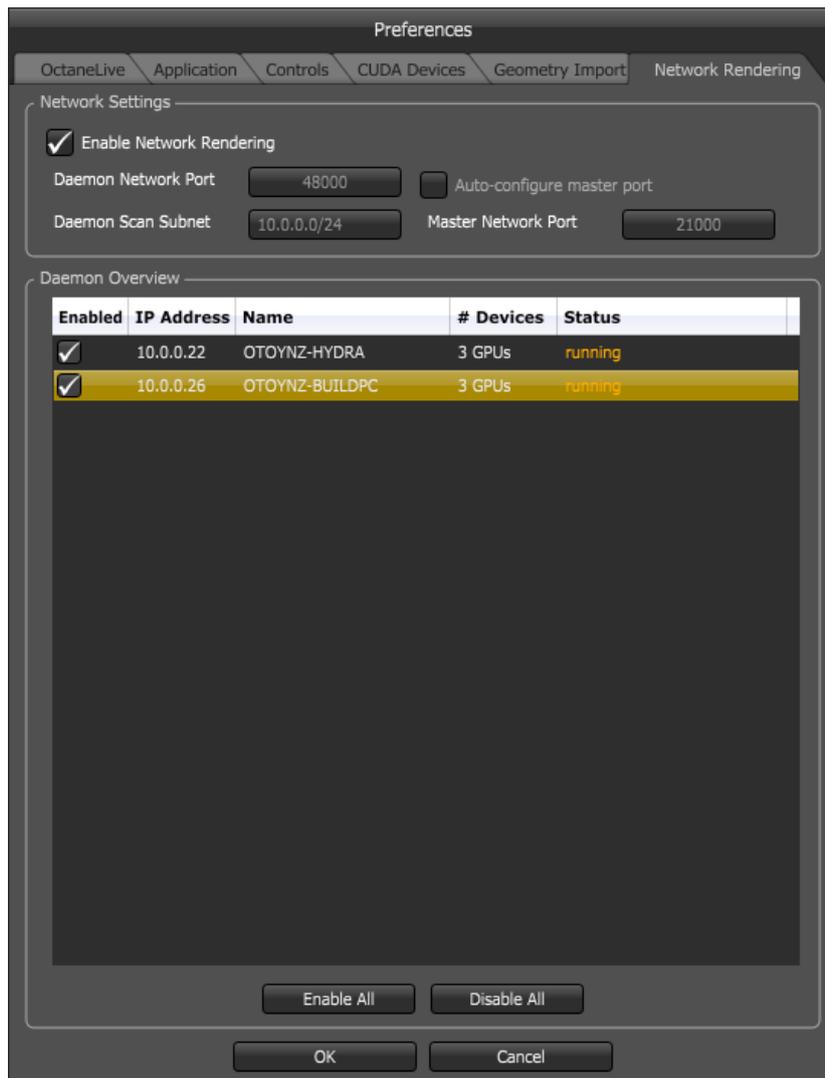
The daemon is in a different subnet than the master is scanning. If you have only one Ethernet adapter on the daemon and master PCs, you can safely ignore this case.

The Windows firewall keeps the master from connecting to the daemon or the daemon from responding to the master. That's the most likely reason. To verify it, disable the firewall for private or home/work networks on both PCs. You have to close the Windows dialog containing the firewall options. Only then the change is actually applied. If the daemon is now detected (should take only up to 10-20 seconds), you can try enabling one firewall after the other to see which one is causing trouble. If you want to have the firewall running, you may have to poke a hole into it, to allow the communication between daemon and master. Only when you enable a daemon in the master settings, the slave gets actually launched and will eventually appear in the status bar of the master. One daemon can be activated only by one master at a time. If daemon is currently "occupied" by another master the user will see the daemon state change accordingly. The automatic port configuration is an option on the master that enables multiple masters to be used on the same computer.

Master

To enable network rendering, go to File -> Preferences -> Network Rendering and tick the box “Enable Network Rendering”.

The network settings dialog includes an option which turns automatic port configuration on the master on or off. If enabled, multiple masters can be used on the same computer.



What then happens is that the master opens a socket at the specified “Master Network Port” and listens at that port for slaves trying to connect to the master. It starts scanning the specified subnet in the local network for daemons and as soon as a slave with the correct version connects, the Octane Standalone status bar will show the additional GPUs and slaves:



The line Launching net render slave (2000002) with master 10.0.0.103:21000 indicates that the slave of version 2000002 successfully connected with the master at IP address 10.0.0.103 on port 21000.

To stop the slave, press CTRL-C and it will gracefully shutdown the process.

Obviously it would be quite painful if users would have to manually start/stop slaves on several computers, which is where the daemons come into the picture.

Daemons

The daemon is a little lightweight program that is launched after login and runs all the time unless manually shutdown. It fulfills various roles:

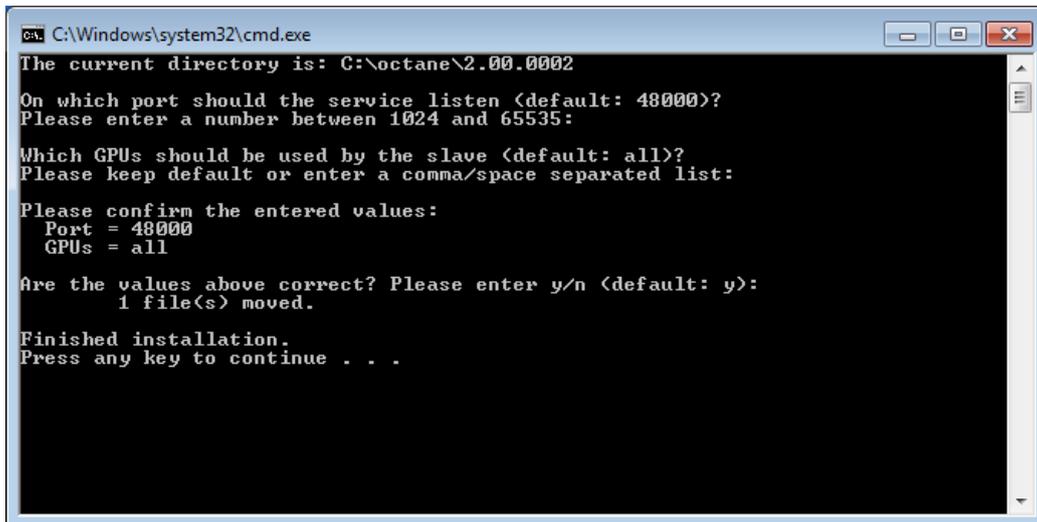
It provides the possibility for masters to locate it in the local network.

It determines the version and GPU configuration of the slave.

It starts/stops a slave on request by a master and makes sure that only one master at a time uses the slave.

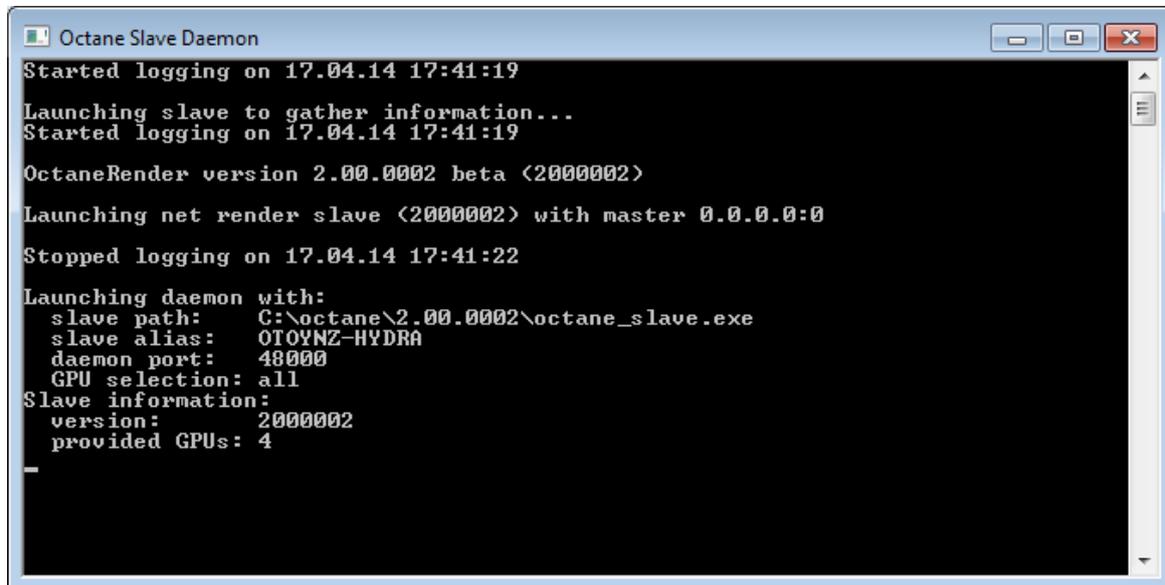
It monitors the “health” of a running slave process and kills the it if necessary.

To set up the daemon, you run the little batch script `_install_daemon.bat` on the slave computer. It will ask you which port the daemon should listen for master requests (if you want to keep the default settings in the brackets, you can just press RETURN):



```
C:\Windows\system32\cmd.exe
The current directory is: C:\octane\2.00.0002
On which port should the service listen (default: 48000)?
Please enter a number between 1024 and 65535:
Which GPUs should be used by the slave (default: all)?
Please keep default or enter a comma/space separated list:
Please confirm the entered values:
  Port = 48000
  GPUs = all
Are the values above correct? Please enter y/n (default: y):
1 file(s) moved.
Finished installation.
Press any key to continue . . .
```

After confirming your selection, a batch file `run_octane_slave_daemon.bat` is created in the startup folder of your start menu (Start -> Programs -> Startup ...). It will be launched the next time you log into your Windows account. A new terminal window will be in your task bar. When you open it, you can see the daemon starting up. At first it tries to launch the slave process to gather some information, which is then displayed:



```
Octane Slave Daemon
Started logging on 17.04.14 17:41:19
Launching slave to gather information...
Started logging on 17.04.14 17:41:19

OctaneRender version 2.00.0002 beta <2000002>
Launching net render slave <2000002> with master 0.0.0.0:0
Stopped logging on 17.04.14 17:41:22

Launching daemon with:
  slave path: C:\octane\2.00.0002\octane_slave.exe
  slave alias: OTOYNZ-HYDRA
  daemon port: 48000
  GPU selection: all
Slave information:
  version: 2000002
  provided GPUs: 4
```