

V-Ray Swarm QuickStart

This page provides a step-by-step introduction to V-Ray Swarm for Rhino.

Introduction

This tutorial covers how to use V-Ray Swarm, a distributed rendering system that allows you to render on multiple machines at once using a simple web interface. Swarm provides dynamic control of multiple machines on your network to render scenes at a significantly faster rate than what can be done locally on a single computer. This tutorial includes how to install Swarm onto machines without even needing Rhino installed, how to access and control the Swarm options, and how to run a distributed render managed by Swarm.

To follow this tutorial, you will need to have the [V-Ray for Rhino plugin installed](#) on a host machine, and at least one other computer to use as a render node. Please note that Swarm can be installed on Windows, Macintosh and Linux computers. In addition, neither Rhino, nor V-Ray for Rhino need to be installed on the secondary machine. This tutorial is a companion to go along with the QuickStart video posted on our YouTube channel and available here:

Tutorial Assets

To download the files used in this tutorial, please click on the button below.



Tutorial Steps

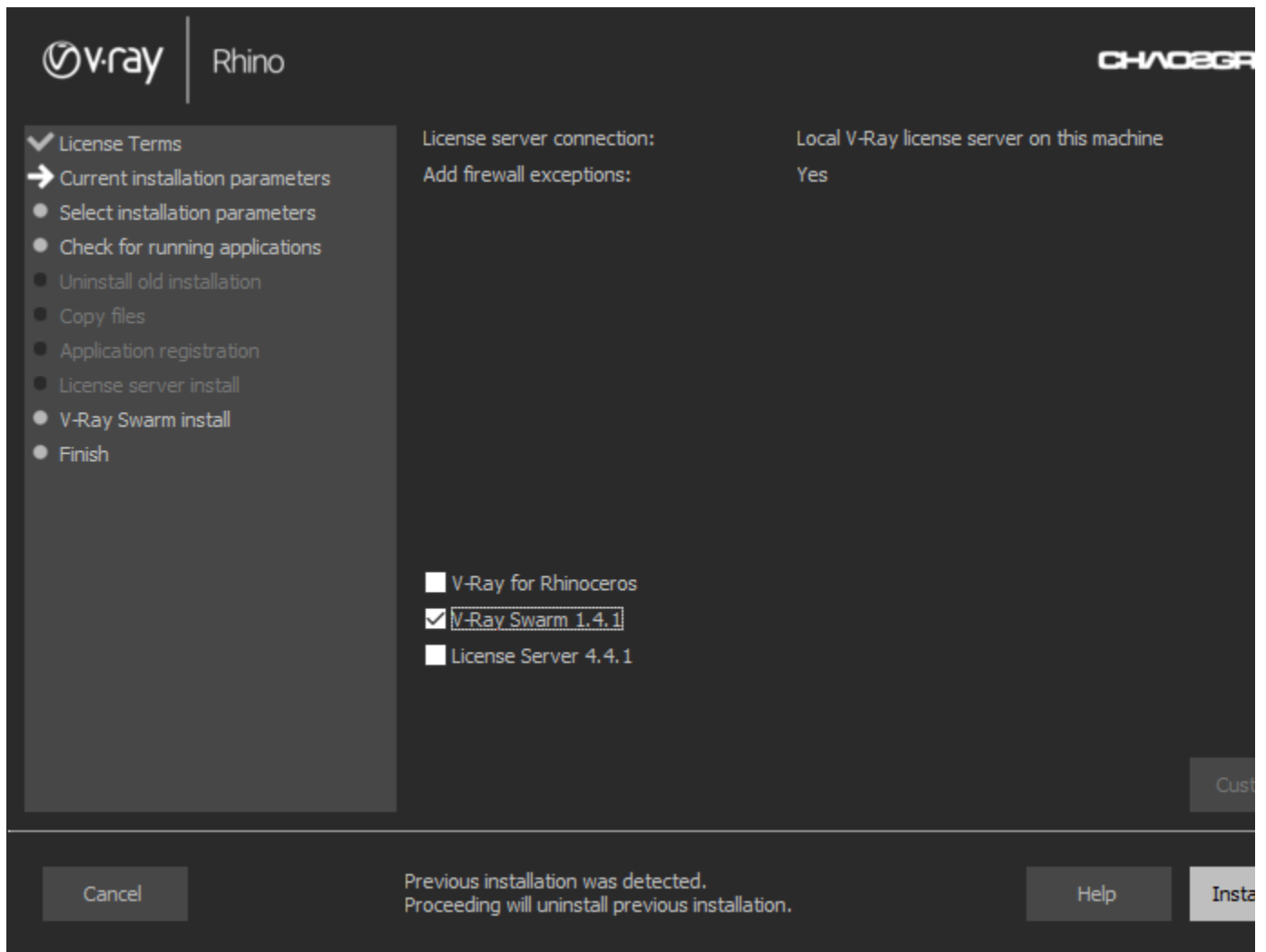
Swarm Installation Options

Swarm can be installed alongside V-Ray for Rhino, but may also be installed on its own by launching the V-Ray installer, and selecting to install only V-Ray Swarm. You do not need to install Rhino or V-Ray for Rhino on a machine that you only want to use as a render node; you can just install Swarm. A standalone V-Ray Swarm installer is available at <https://download.chaosgroup.com/?product=58>.

Likewise, you don't need to install Swarm if you do not want the machine to be a render node. You can just install Rhino and V-Ray for Rhino without Swarm and still be able to render on render nodes with Swarm.

If you already have V-Ray for Rhino installed and licensed on a machine on which you wish to install Swarm, you will not need to install the license server. However, if you are installing Swarm on a machine that does not have V-Ray for Rhino already installed and licensed, you will need to click the "Customize" button and in the follow-up menu fill in the IP of the machine where your Online License Server is running. For general instructions on V-Ray and Swarm setup and installation, [please click here](#).

For this tutorial, we install just Swarm on a render node machine that already has the license server installed. Run the Installer and only check **V-Ray Swarm 1.4.1** to install that component.



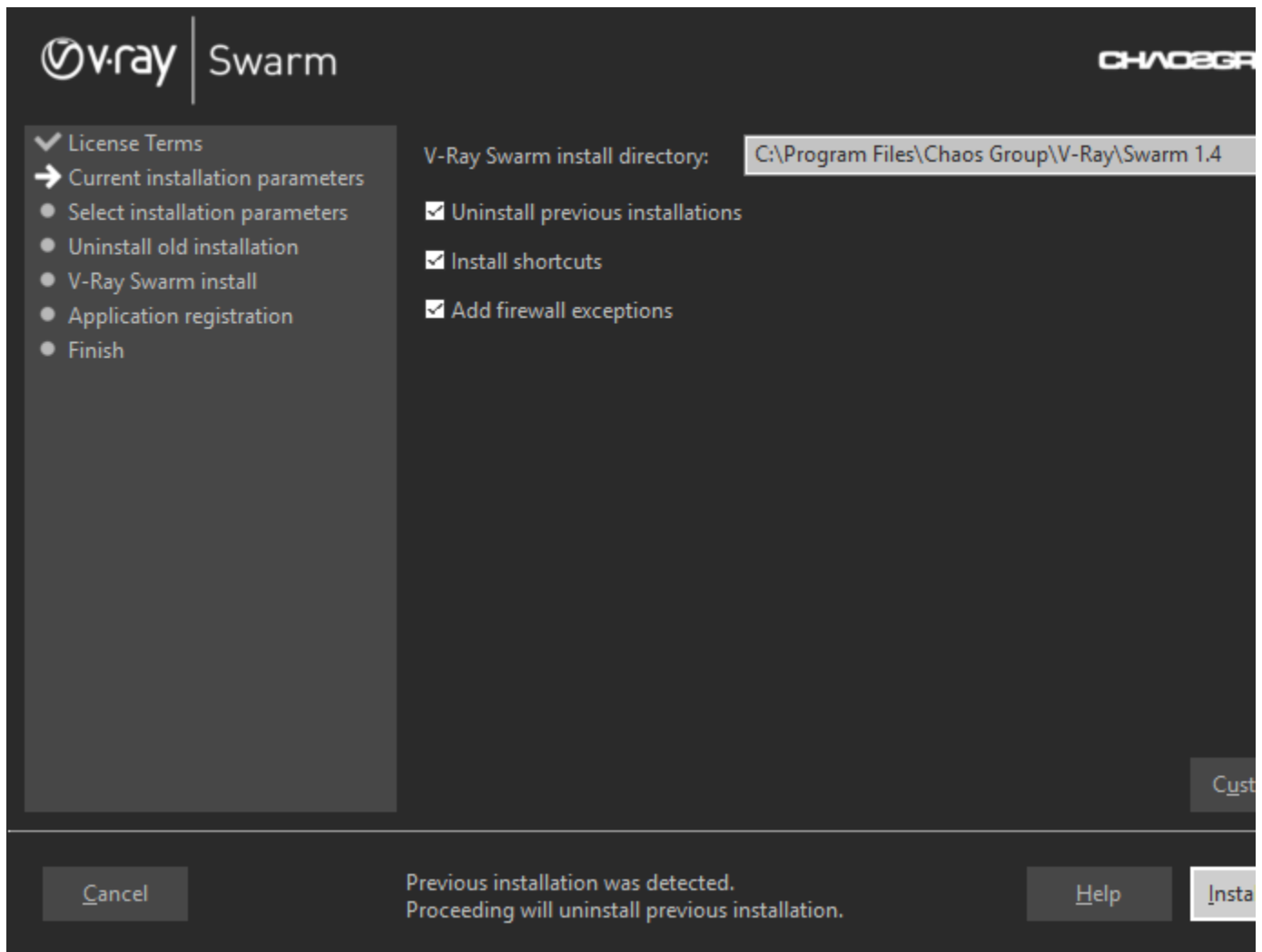
NOTE:

You likely don't need to change any of the settings. Just make sure to uninstall any previous installations (if you have any), and make sure **Add firewall exceptions** is checked on. Click **Install Now**.

That's it. You'll need to install Swarm on all the machines to which you wish to distribute renders.

The local machine always uses 1 render node license during rendering. A render node license is required for each additional render node machine participating in the job.

V-Ray 3 for Rhino trials come with 1 render node license. To evaluate V-Ray Swarm, you can request additional render nodes by using the following form.

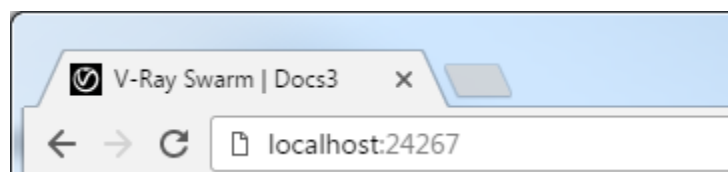


Swarm Configuration

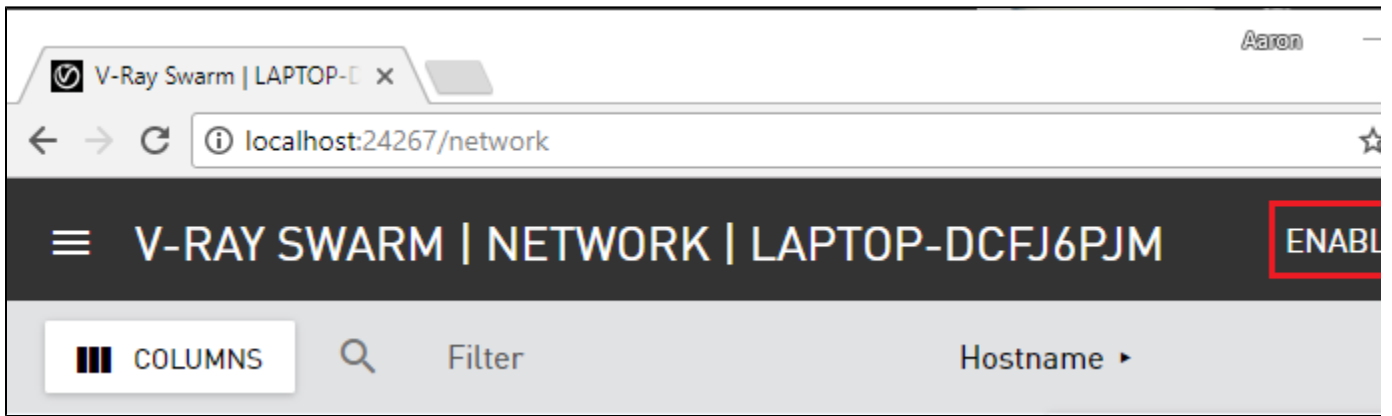
Now that Swarm is installed, let's get into the configuration interface for the node.

Open up a web browser. To access the interface, you'll need to type in the browser URL field "`localhost:24267`" as shown below. 24267 is the default port used for Swarm. If you changed the port during installation, enter that port number instead.


As a side note, you can remotely access Swarm installed on a different machine by typing in that machine's IP address and the port number instead of using "local host".

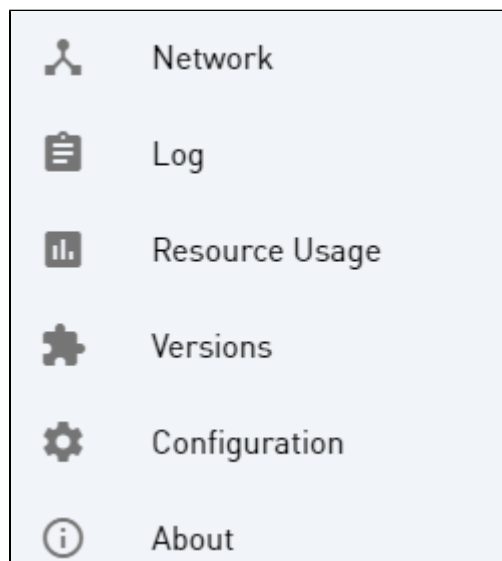


The Swarm interface greets you with the **Network** page, and we can see the name of the current machine up at the top, as well as the other machine names in the list. In the upper-right corner you can enable or disable the node for Swarm rendering.



Swarm Menu

Click the  icon in the upper-left corner to access the different pages for the interface, listed below.




Log Page

Switch to the **Log** page to see a log of messages from this machine's render job progress, which is good for troubleshooting, if it's needed.

≡ V-RAY SWARM | LOG | LAPTOP-DCFJ6PJM

```
[2017/Aug/18|20:28:03] Number of intersectable primitives: 512176
[2017/Aug/18|20:28:03] SD triangles: 346363
[2017/Aug/18|20:28:03] MB triangles: 0
[2017/Aug/18|20:28:03] Static primitives: 165812
[2017/Aug/18|20:28:03] Moving primitives: 0
[2017/Aug/18|20:28:03] Infinite primitives: 1
[2017/Aug/18|20:28:03] Static hair segments: 0
[2017/Aug/18|20:28:03] Moving hair segments: 0
[2017/Aug/18|20:28:03] Cleaning up bitmap manager
[2017/Aug/18|20:28:03] Stopping V-Ray...
[2017/Aug/18|20:28:03] Closing DR session from 192.168.48.102
[2017/Aug/18|20:28:03] V-Ray process finished
[2017/Aug/18|20:28:03] Reading configuration from "C:\Program Files\Chaos Group\V-Ray\Swarm 1.4\work\
Ray\06086a6f-db47-4f2e-bcac-8b337f59c9ce\bin\vrayconfig.xml"
[2017/Aug/18|20:28:03] =====
[2017/Aug/18|20:28:03] V-Ray Standalone, version 3.60.03 for x64
[2017/Aug/18|20:28:03] Copyright (C) 2000-2017 Chaos Group Ltd. All rights reserved.
[2017/Aug/18|20:28:03] Use -credits option for additional third-party copyright notices.
[2017/Aug/18|20:28:03]
[2017/Aug/18|20:28:03] Build from Aug 9 2017, 15:43:29
[2017/Aug/18|20:28:03] Compiled with Intel C++ compiler, version 16
[2017/Aug/18|20:28:03] Operating system is Microsoft(tm) Windows(tm), version 6.2,
[2017/Aug/18|20:28:03]
[2017/Aug/18|20:28:03] V-Ray core version is 3.60.03
[2017/Aug/18|20:28:03] =====
[2017/Aug/18|20:28:03]
[2017/Aug/18|20:28:04] Loading plugins from "C:\Program Files\Chaos Group\V-Ray\Swarm 1.4\work\vray-
db47-4f2e-bcac-8b337f59c9ce\bin\plugins\vray_*.dll"
```

The () icon accesses display options for the log, as shown below.

≡ V-RAY SWARM | LOG | LAPTOP-DCFJ6PJM

[2017/Aug/18|20:28:03] Number of intersectable primitives: 512176

[2017/Aug/18|20:28:03] SD triangles: 346363

[2017/Aug/18|20:28:03] MB triangles: 0

[2017/Aug/18|20:28:03] Static primitives: 165812

[2017/Aug/18|20:28:03] Moving primitives: 0

[2017/Aug/18|20:28:03] Infinite primitives: 1

[2017/Aug/18|20:28:03] Static hair segments: 0

[2017/Aug/18|20:28:03] Moving hair segments: 0

[2017/Aug/18|20:28:03] Cleaning up bitmap manager

[2017/Aug/18|20:28:03] Static primitives: 165812

Resource Usage Page

In the Swarm menu, switch to the **Resource Usage** page. This shows the current usage of different aspects of the node, including CPU, GPU, memory, network traffic, and storage space available.

CPU

GPU

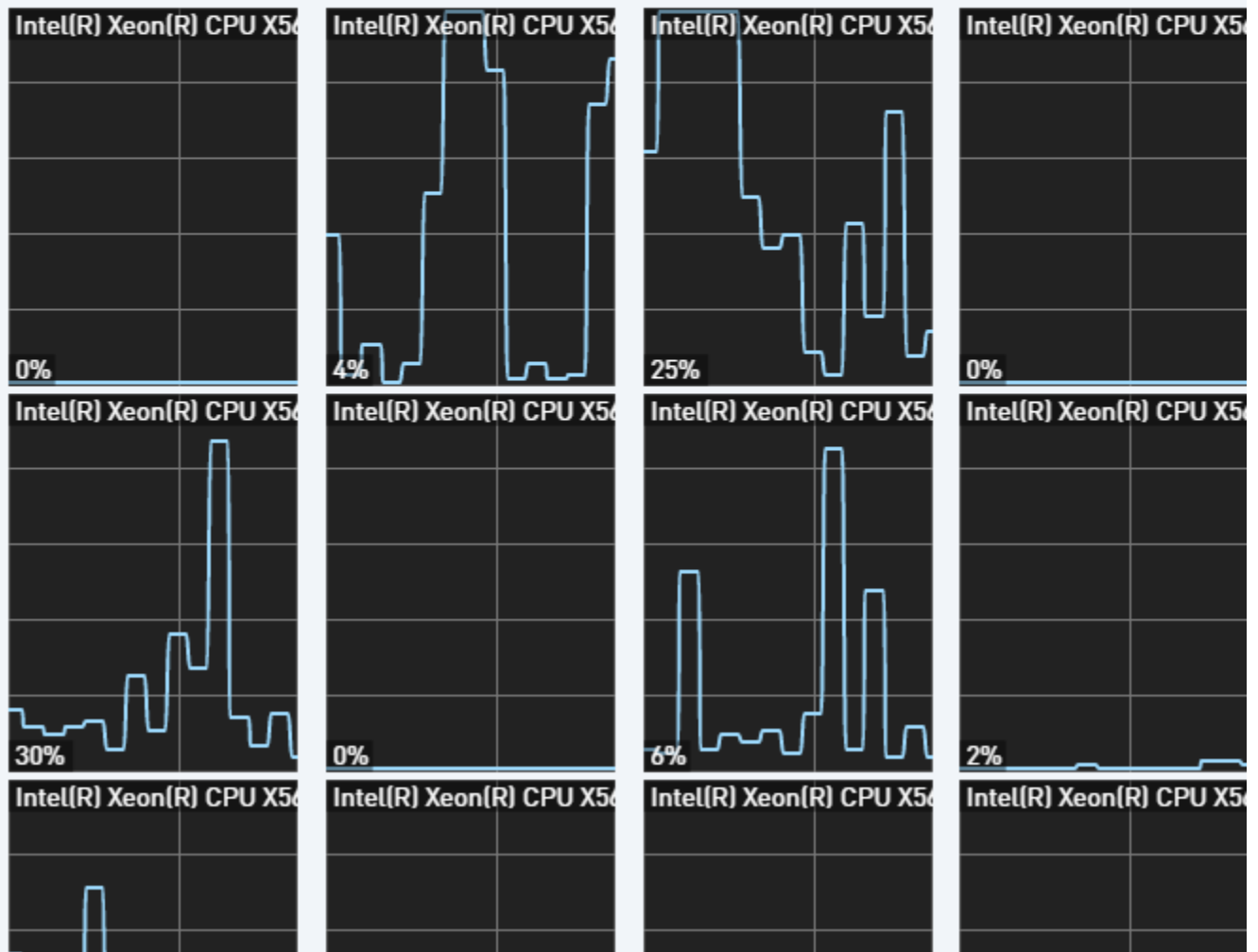
MEMORY

NETWORK

STORAGE

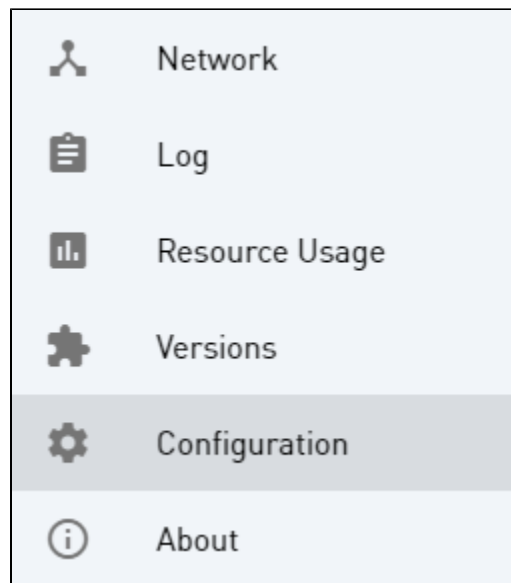
Columns

4

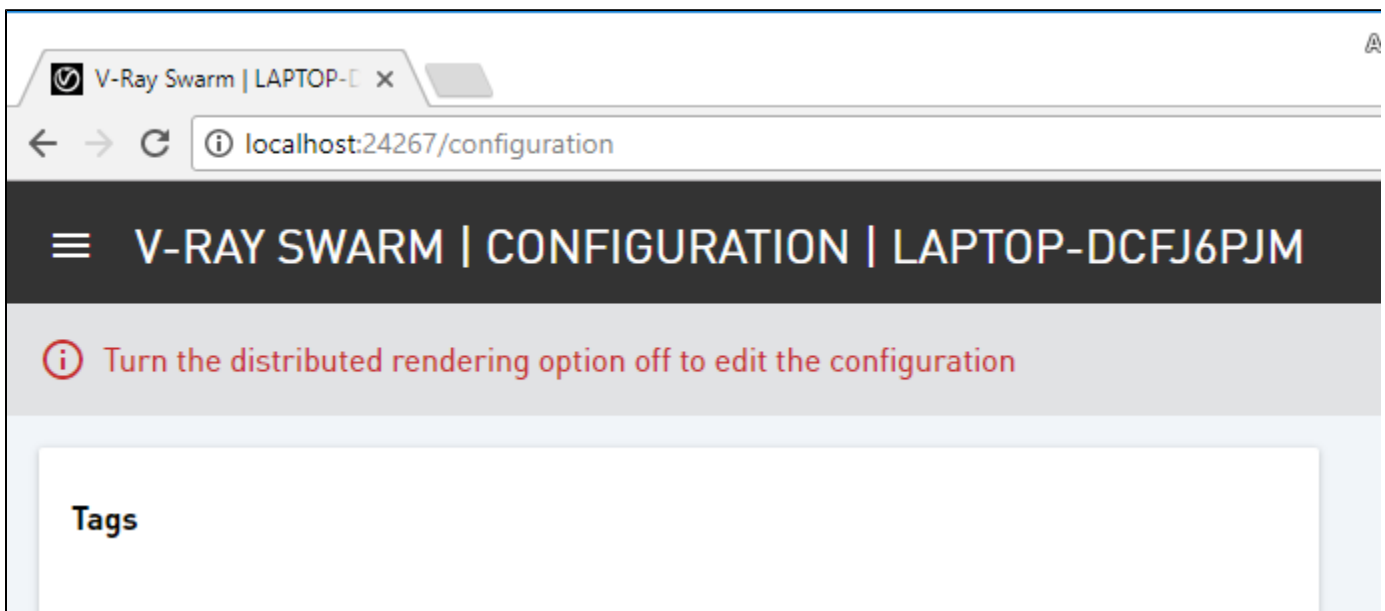


Configuration Page

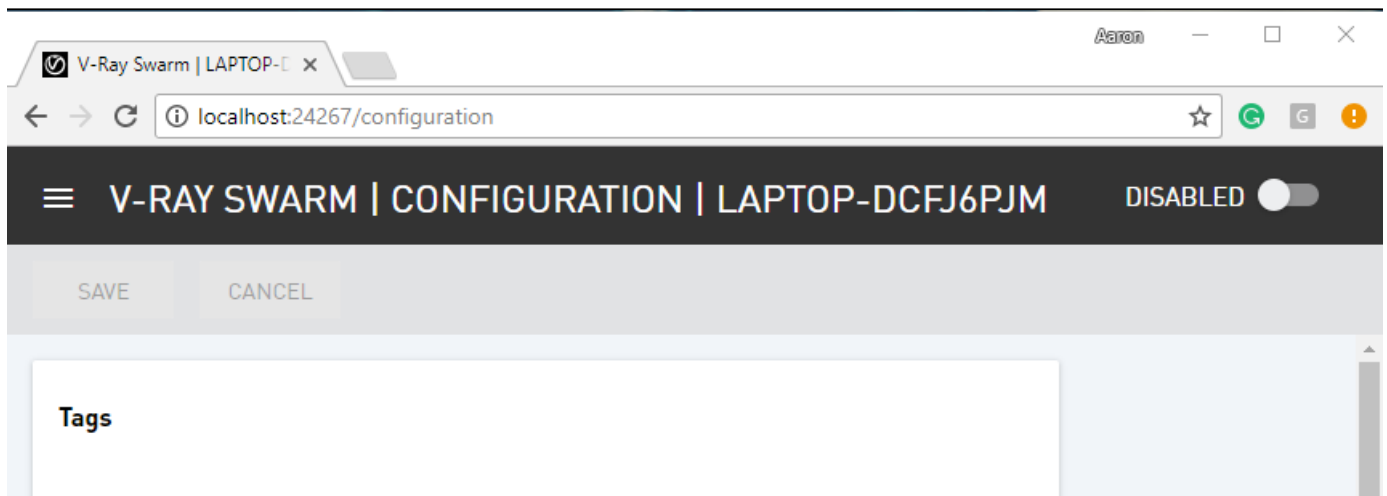
Now switch to the **Configuration** page from the Swarm menu.



Making any changes to Configuration requires that the node first be *disabled* for rendering.



Disable the node through the icon at the top right, and the options will be available to modify.



Tags allow you to define this node with a descriptor that defines something about the machine, which we can later use to easily select which machines we want to render from inside of Rhino.

For example, if this machine is one of a few that have a powerful GPU, or if a machine is not a workstation and just a node on a render farm.

All nodes have the **default** tag by default.

Let's tag this machine as *render_node* and click **Add**.

A screenshot of the 'Tags' configuration form. It has a title 'Tags' and a text input field labeled 'Add new tag' containing the text 'render_node'. To the right of the input field is a gray button labeled 'ADD', which is highlighted with a red rectangular border. Below the input field, there is a 'Default' button and an information icon (i) followed by the text 'At least one tag is required'.

Then click **Save** to save all changes to the node.

≡ V-RAY SWARM | CONFIGURATION | LAPTOP-DCFJ6PJM

SAVE

CANCEL

Tags

Add new tag

ADD

Default × render_node ×

Now, you can re-enable the node.

Network Configuration

Now let's take a look at the **Network** page.

This page shows all the available Swarm machines on the network, displaying their machine names, status, IP address and their tags right here, along with some important information like the V-Ray version. Here we confirm the addition of the *render_node* tag that we added to this machine, LAPTOP-DCFJ6PJM.

☰

V-RAY SWARM | NETWORK | LAPTOP-DCFJ6PJM

☐ COLUMNS

🔍 Filter

Hostname ▾

<div><input type="checkbox"/> ▲</div> HOSTNAME	STATUS	IP ADDRESS	TAGS	V-RAY VERSION	PLATFORM	CPU(S)	GPU(S)	CPU	MEMORY
<input checked="" type="checkbox"/> CGUSA-050	DISABLED	192.168.48.107	Default	3.52.03 51b9cc1	Windows	32	1	0%	6%
<input type="checkbox"/> CGUSA-052	ENABLED	192.168.48.102	Default, render_node	3.40.04 75d98d7	Windows	16	4	5%	12%
<input type="checkbox"/> CGUSA-053	ENABLED	192.168.48.48	Default	3.60.03 6c2bd6a	Windows	12	2	0%	13%
<input checked="" type="checkbox"/> CGUSA-51	DISABLED	192.168.48.135	Default	3.52.03 51b9cc1	Windows	32		0%	5%
<input type="checkbox"/> DESKTOP-PD7TKGK	ENABLED	192.168.48.49	Default, Ted-OLD	3.57.01 f18e02b	Windows	8	5	1%	10%
<input type="checkbox"/> ★ fpdrogo-PC	ENABLED	192.168.48.90	Default, Fer	3.40.04 75d98d7	Windows	16	3	7%	36%
<input type="checkbox"/> GPUBOX02	ENABLED	192.168.48.69	GPU	3.40.04 75d98d7	Windows	12	4	0%	11%
<input type="checkbox"/> LAPTOP-DCFJ6PJM	ENABLED	192.168.48.165	Default, render_node	3.60.03 6c2bd6a	Windows	8	2	14%	61%
<input type="checkbox"/> renderPRO	ENABLED	192.168.48.91	RenderPro	3.60.03 6c2bd6a	Windows	56		0%	10%

Through this page, you can select a machine and remotely enable or disable it through the information panel. We can select multiple machines using the Shift and/or Control keys when selecting additional nodes.

cgusademo-PC

Actions



Enable



Disable



Copy Configuration

Node information

ENABLED

Status

Copy Configuration

Copy Configuration will duplicate the settings from your current machine to the one selected in the list. You can also copy settings to multiple nodes at once by selecting them with the Shift key or the Ctrl key. So I'll go ahead and copy my configuration to **CGUSA-50** and **CGUSA-51** by first selecting those machines in the list.

☰

V-RAY SWARM | NETWORK | LAPTOP-DCFJ6PJM

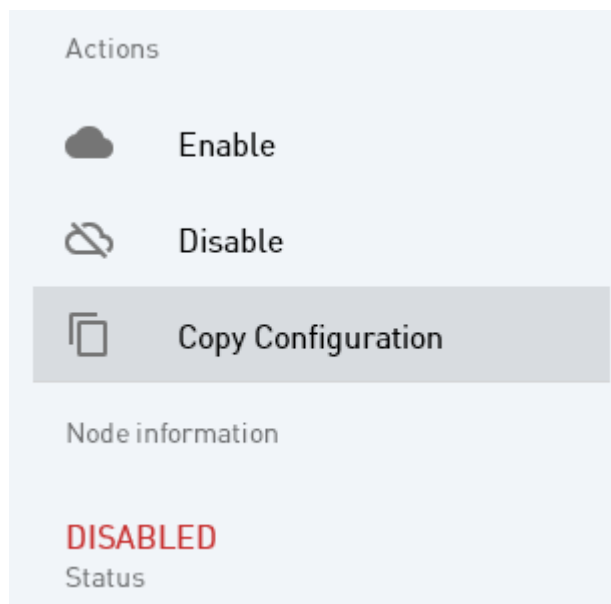
COLUMNS

Filter

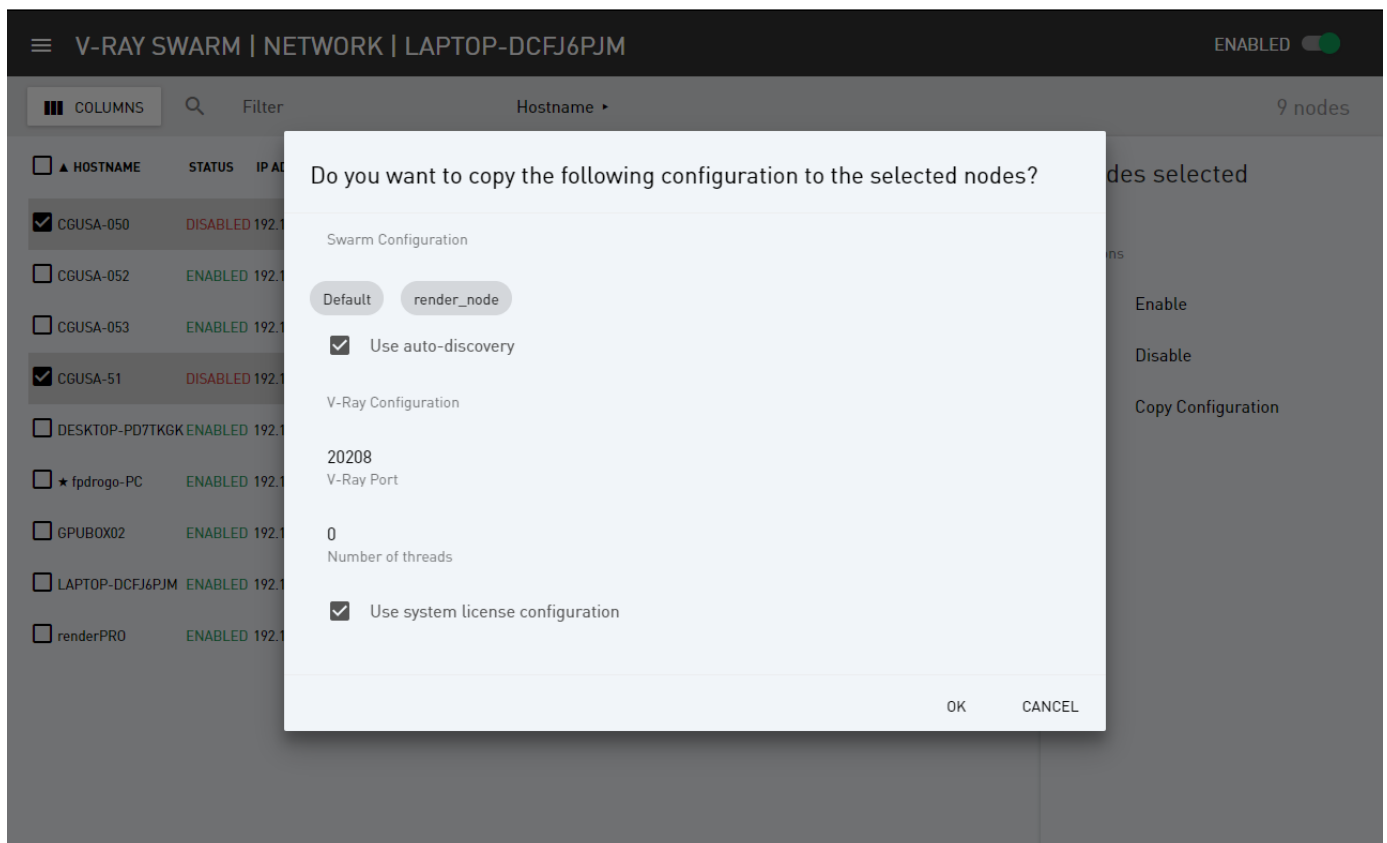
Hostname ▶

<input type="checkbox"/> ▲ HOSTNAME	STATUS	IP ADDRESS	TAGS	V-RAY VERSION	PLATFORM	CPU(S)
<input checked="" type="checkbox"/> CGUSA-050	DISABLED	192.168.48.107	Default	3.52.03 51b9cc1	Windows	32
<input type="checkbox"/> CGUSA-052	ENABLED	192.168.48.102	Default, render_node	3.40.04 75d98d7	Windows	16
<input type="checkbox"/> CGUSA-053	ENABLED	192.168.48.48	Default	3.60.03 6c2bd6a	Windows	12
<input checked="" type="checkbox"/> CGUSA-51	DISABLED	192.168.48.135	Default	3.52.03 51b9cc1	Windows	32
<input type="checkbox"/> DESKTOP-PD7TKGK	ENABLED	192.168.48.49	Default, Ted-OLD	3.57.01 f18e02b	Windows	8
<input type="checkbox"/> ★ fpdrogo-PC	ENABLED	192.168.48.90	Default, Fer	3.40.04 75d98d7	Windows	16
<input type="checkbox"/> GPUBOX02	ENABLED	192.168.48.69	GPU	3.40.04 75d98d7	Windows	12
<input type="checkbox"/> LAPTOP-DCFJ6PJM	ENABLED	192.168.48.165	Default, render_node	3.60.03 6c2bd6a	Windows	8
<input type="checkbox"/> renderPRO	ENABLED	192.168.48.91	RenderPro	3.60.03 6c2bd6a	Windows	56

First **disable** them, then click **Copy Configuration**.



This copies the configuration from the current machine (the one you have currently accessed through your web browser; in this case, LAPTOP-DCFJ6PJM) to the two machines that are selected.



You can verify that the settings have copied over by taking note that the tags have carried over to the two machines. Let's re-enable them to bring them back online.

≡

V-RAY SWARM | NETWORK | LAPTOP-DCFJ6PJM

COLUMNS

🔍

Filter

Hostname ▾

<input type="checkbox"/> ▲ HOSTNAME	STATUS	IP ADDRESS	TAGS	V-RAY VERSION	PLATFORM	CPU(S)	GPU(S)	CPU	MEMORY
<input checked="" type="checkbox"/> CGUSA-050	ENABLED	192.168.48.107	Default, <u>render_node</u>	3.52.03 51b9cc1	Windows	32	1	0%	6%
<input type="checkbox"/> CGUSA-052	ENABLED	192.168.48.102	Default, render_node	3.40.04 75d98d7	Windows	16	4	4%	12%
<input type="checkbox"/> CGUSA-053	ENABLED	192.168.48.48	Default	3.60.03 6c2bd6a	Windows	12	2	0%	13%
<input checked="" type="checkbox"/> CGUSA-51	ENABLED	192.168.48.135	Default, <u>render_node</u>	3.52.03 51b9cc1	Windows	32		1%	5%
<input type="checkbox"/> DESKTOP-PD7TKGK	ENABLED	192.168.48.49	Default, Ted-OLD	3.57.01 f18e02b	Windows	8	5	1%	10%
<input type="checkbox"/> ★ fpdrogo-PC	ENABLED	192.168.48.90	Default, Fer	3.40.04 75d98d7	Windows	16	3	94%	36%
<input type="checkbox"/> GPUBOX02	ENABLED	192.168.48.69	GPU	3.40.04 75d98d7	Windows	12	4	1%	11%
<input type="checkbox"/> LAPTOP-DCFJ6PJM	ENABLED	192.168.48.165	Default, render_node	3.60.03 6c2bd6a	Windows	8	2	24%	61%
<input type="checkbox"/> renderPRO	ENABLED	192.168.48.91	RenderPro	3.60.03 6c2bd6a	Windows	56		0%	10%

You can also log into other machines on the network from the Network page. If you select a node, you can access that machine's **Swarm** interface by clicking where it gives the URL here in the interface, shown below.

CGUSA-050

Actions



Enable



Disable



Copy Configuration

Node information

ENABLED

Status

<http://192.168.48.107:24267/>

Url

3.52.03 51b9cc1 Jun 19 2017
18:00:55

V-Ray Version

Windows

Platform

Tags

Default

render_node

This opens a new tab, showing the user interface for that machine, so managing several nodes can be much easier than going to each physical machine.

V-Ray Swarm | LAPTOP-DC x

V-Ray Swarm | CGUSA-05 x

192.168.48.107:24267/network

≡

V-RAY SWARM | NETWORK | CGUSA-050

COLUMNS

Filter

Hostname

▾

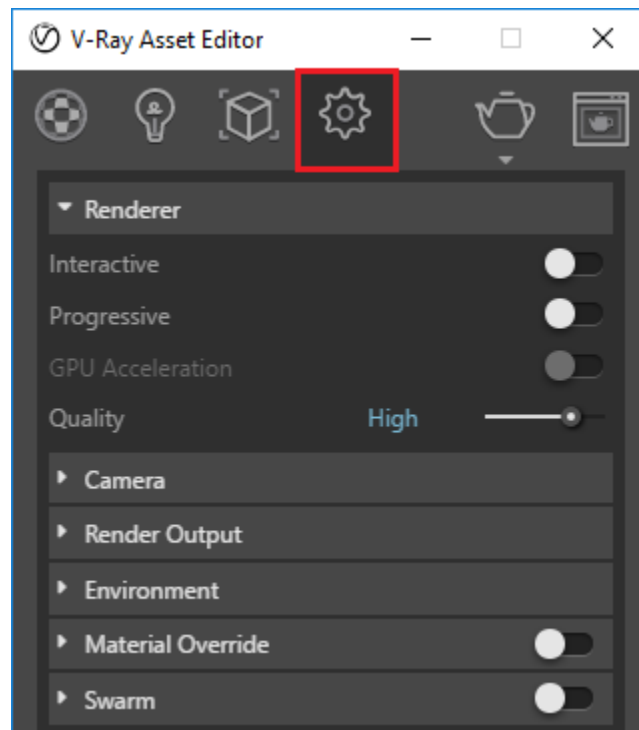
<input type="checkbox"/>	▲ HOSTNAME	STATUS	IP ADDRESS	TAGS	V-RAY VERSION
<input type="checkbox"/>	CGUSA-050	ENABLED	192.168.48.107	Default, render_node	3.52.03 51b9cc1
<input type="checkbox"/>	CGUSA-052	ENABLED	192.168.48.102	Default, render_node	3.40.04 75d98d7
<input type="checkbox"/>	CGUSA-053	ENABLED	192.168.48.48	Default	3.60.03 6c2bd6a
<input type="checkbox"/>	CGUSA-51	ENABLED	192.168.48.135	Default, render_node	3.52.03 51b9cc1
<input type="checkbox"/>	DESKTOP-PD7TKGK	ENABLED	192.168.48.49	Default, Ted-OLD	3.57.01 f18e02b
<input type="checkbox"/>	★ fpdrogo-PC	ENABLED	192.168.48.90	Default, Fer	3.40.04 75d98d7
<input type="checkbox"/>	GPUBOX02	ENABLED	192.168.48.69	GPU	3.40.04 75d98d7
<input type="checkbox"/>	LAPTOP-DCFJ6PJM	ENABLED	192.168.48.165	Default, render_node	3.60.03 6c2bd6a
<input type="checkbox"/>	renderPRO	ENABLED	192.168.48.91	RenderPro	3.60.03 6c2bd6a

Setting up V-ray for Rhino to use the Swarm

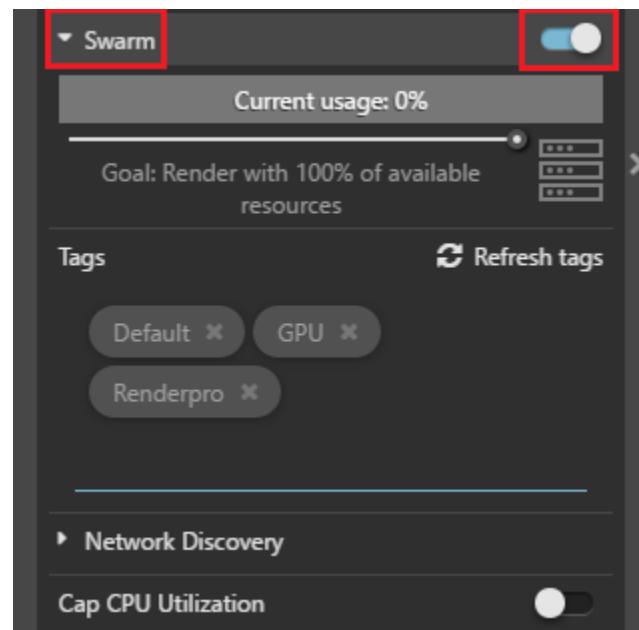
Now let's see Swarm in action on a Rhino project.

Open up any project. Here we're using **Swarm_Start.3dm** which you can download from the link at the top of this page, or use a sample project or a personal project of your own.

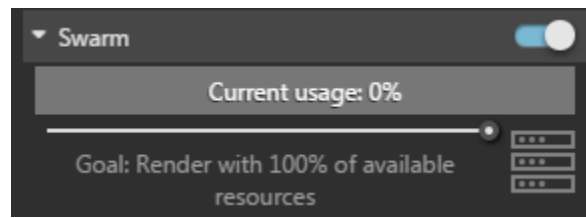
Let's go over the settings for Swarm. Once you have the project loaded, open the **Asset Editor** from the V-Ray toolbar. Click the **Settings** tab.



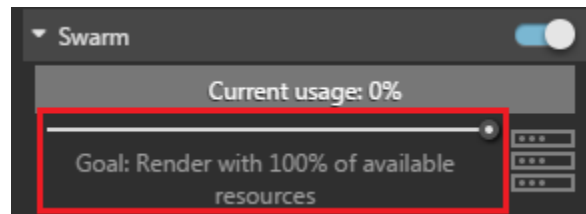
Expand the **Swarm** roll out and enable Swarm () and you'll see the full set of **Swarm** settings.



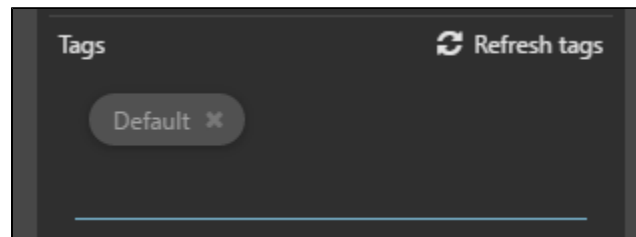
The first setting is **Current usage**, which shows the percentage of utilization of available resources.



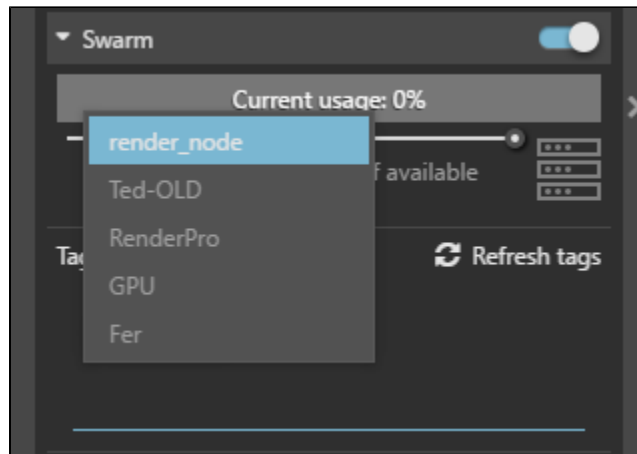
The **Goal** slider below determines how much of the available Swarm resources to throw at the rendering, and is defined by a percentage of the available machines. If you slide the **Goal** slider to the right, you can dynamically add more machines to this render. Computers added or removed to V-Ray Swarm will be factored into the percentage automatically.



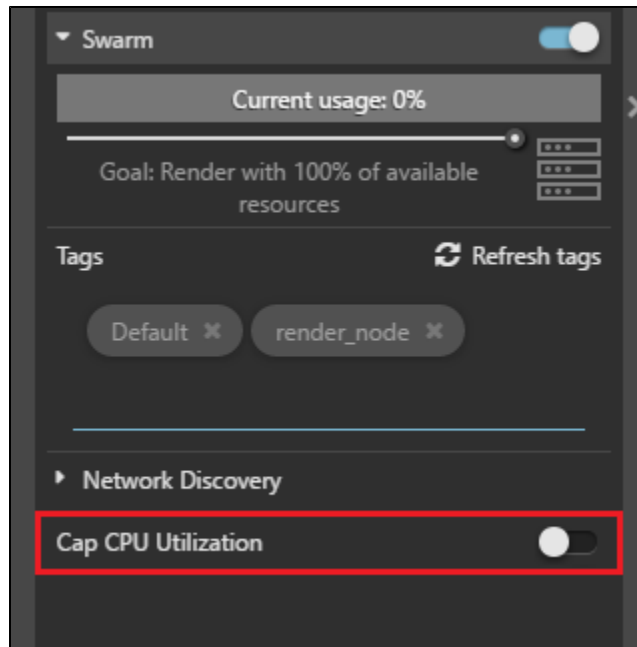
Tags is where the *render_node* tag we created comes in handy. V-Ray for Rhino will only render to machines that have any one of the tags listed here. You can click on the **X** next to the tags to remove them from the list. When no Tags are chosen, all available render node machines visible in the Swarm manager will be used.



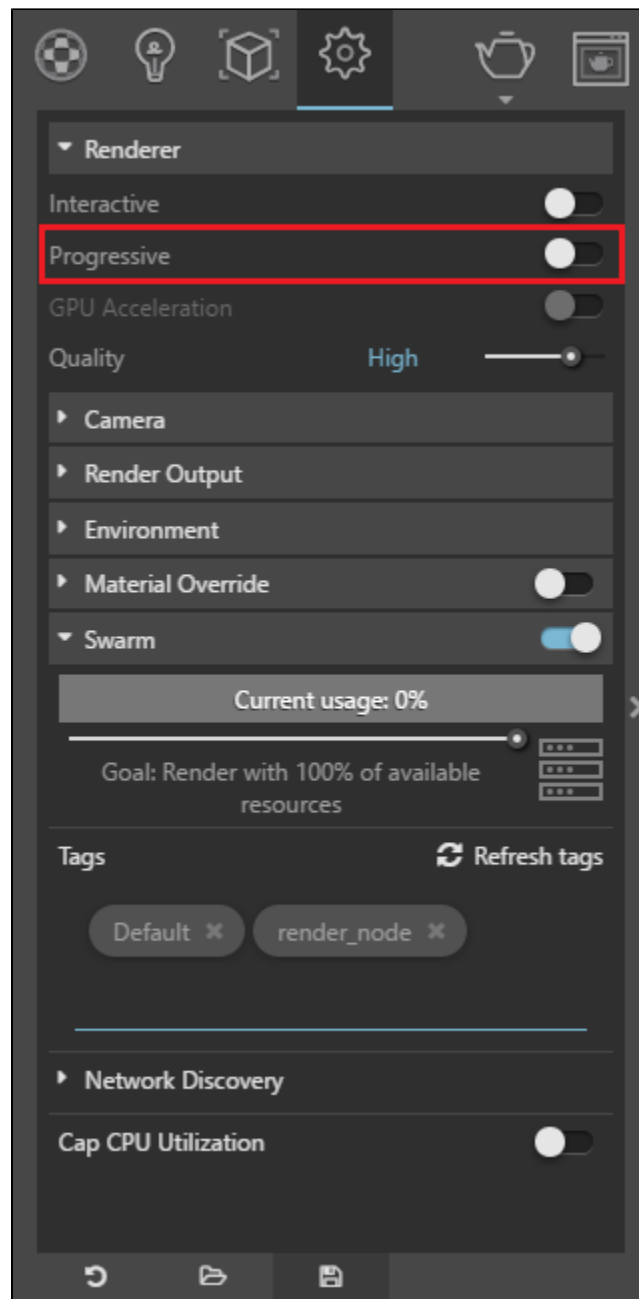
Click on the text field immediately below the *Default* tag icon (shown above) to add tags as needed by typing them in, or selecting from the drop-down menu. Add the *render_node* tag.



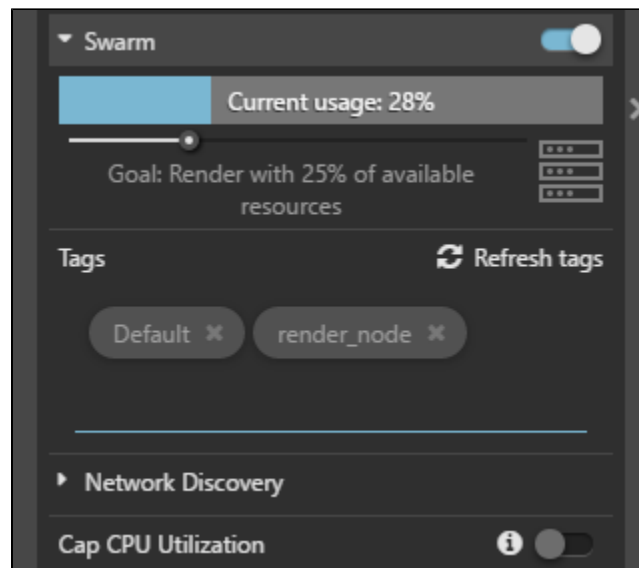
By enabling the **Cap CPU Utilization** button, you'll cap the CPU usage of the local machine to a single core. Let's leave it off for this tutorial and the local machine will render at full capacity.



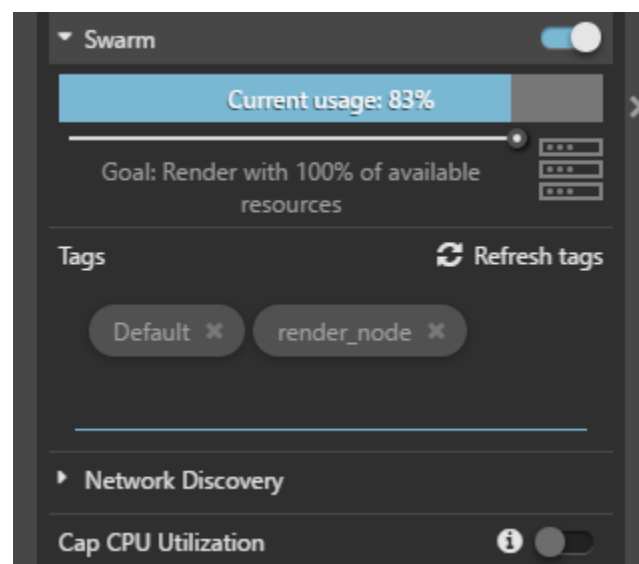
All the render settings are taken care of in this scene to output the final image. As you can see, **Progressive** is disabled, meaning V-Ray for Rhino will use **Buckets** for the rendering. This is typically better when you want to use Swarm, as distributing the render is more optimized when using **Buckets** over **Progressive**.



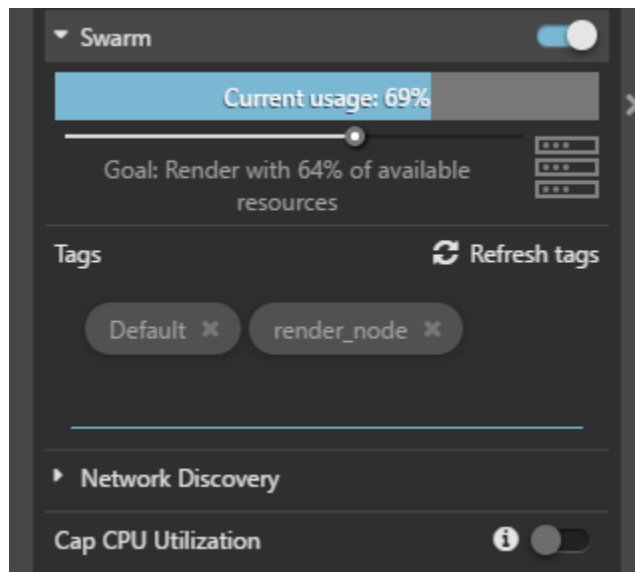
And since **Swarm** is already enabled, go ahead and start a render. After a moment the render begins and you'll notice the usage stops at 33%. Once it got past 25% it stopped adding render nodes to the Swarm.



But as it's rendering, you can increase the **Goal** slider to dynamically add more resources to the render.



And of course, you can decrease the goal as well, and the nodes will drop off the Swarm.



Now I'll add a couple more tags to the Swarm pool, and more nodes will populate the render.




Zoom into the render and you can see each node name in the individual buckets of the render to see which node is rendering which bucket.



Managing the swarm while rendering



Click () icon, and the Swarm interface of the currently elected Coordinator opens in a web browser, and you can see a more detailed view of what the Swarm is doing on this render.

Remove resources from the render, and you can see the interface update as those machines drop off. Their status will change from **RENDERING** to **ENABLED**.

V-RAY SWARM | NETWORK | LAPTOP-DCFJ6PJM

COLUMNS



Filter

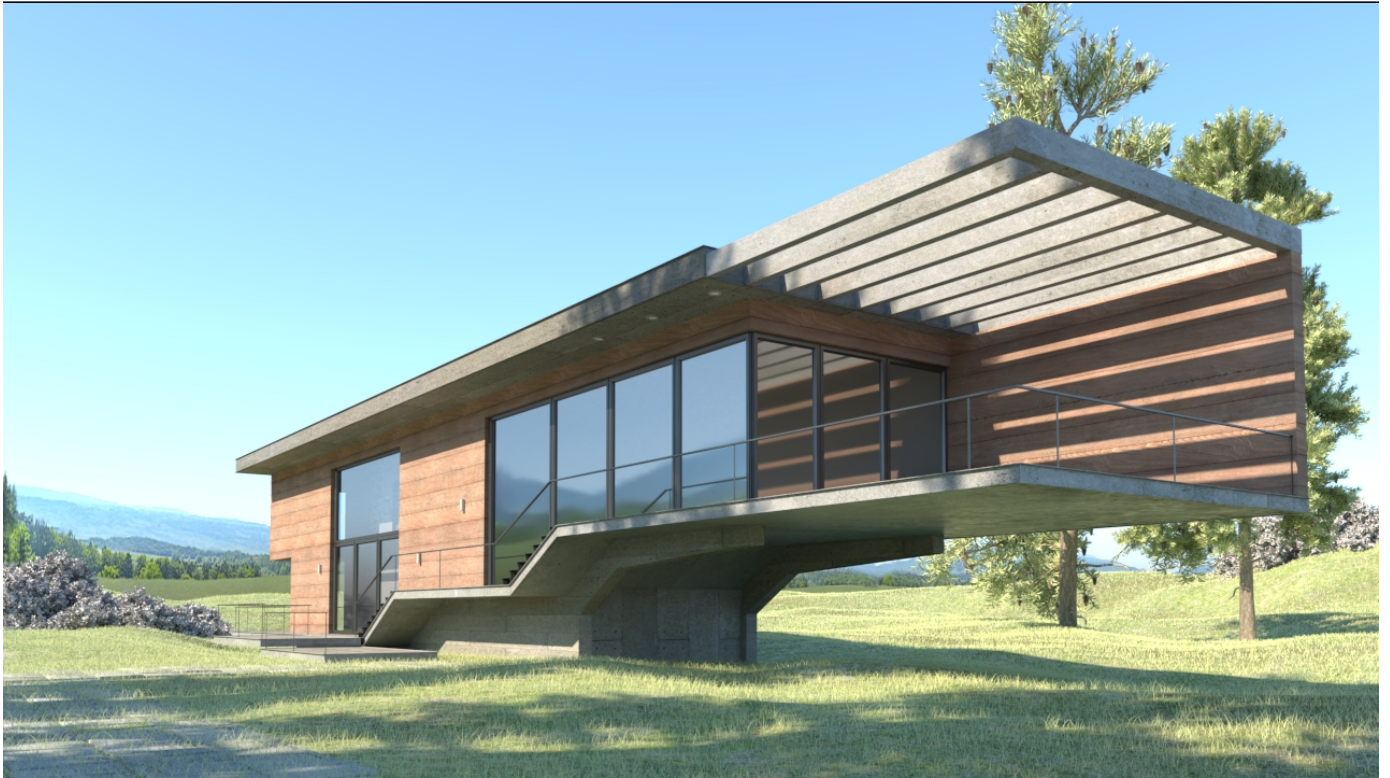
Hostname ▾

<div> <div></div> <div>▲ HOSTNAME</div> </div>	STATUS	IP ADDRESS	TAGS	V-RA
<div> <div></div> <div>CGUSA-050</div> </div>	RENDERING (192.168.48.165)	192.168.48.107	Default, render_node	3.40.0
<div> <div></div> <div>CGUSA-052</div> </div>	RENDERING (192.168.48.165)	192.168.48.102	Default, render_node	3.40.0
<div> <div></div> <div>CGUSA-053</div> </div>	ENABLED	192.168.48.48	Default	3.40.0
<div> <div></div> <div>CGUSA-51</div> </div>	RENDERING (192.168.48.165)	192.168.48.135	Default, render_node	3.40.0
<div> <div></div> <div>DESKTOP-PD7TKGK</div> </div>	ENABLED	192.168.48.49	Default, Ted-OLD	3.40.0
<div> <div></div> <div>fpdrogo-PC</div> </div>	ENABLED	192.168.48.90	Default, Fer	3.40.0
<div> <div></div> <div>GPUBOX02</div> </div>	RENDERING (192.168.48.165)	192.168.48.69	GPU	3.40.0
<div> <div></div> <div>LAPTOP-DCFJ6PJM</div> </div>	ENABLED	192.168.48.165	Default, render_node	3.60.0
<div> <div></div> <div>★ LENOVOVR</div> </div>	ENABLED	192.168.48.160	Default	3.40.0
<div> <div></div> <div>renderPRO</div> </div>	RENDERING (192.168.48.165)	192.168.48.91	RenderPro	3.40.0

Render Time Comparison

Here is the completed render with a time comparison of using just the local machine against using V-Ray Swarm, saving a lot of time over rendering on just the local machine.

Swarm rendering is most time-efficient when used for jobs with long base render time.



Host machine render time (8 cores) : 4 m 47.8 s
Swarm render time (48 Cores) : 1 m 56.4 s