

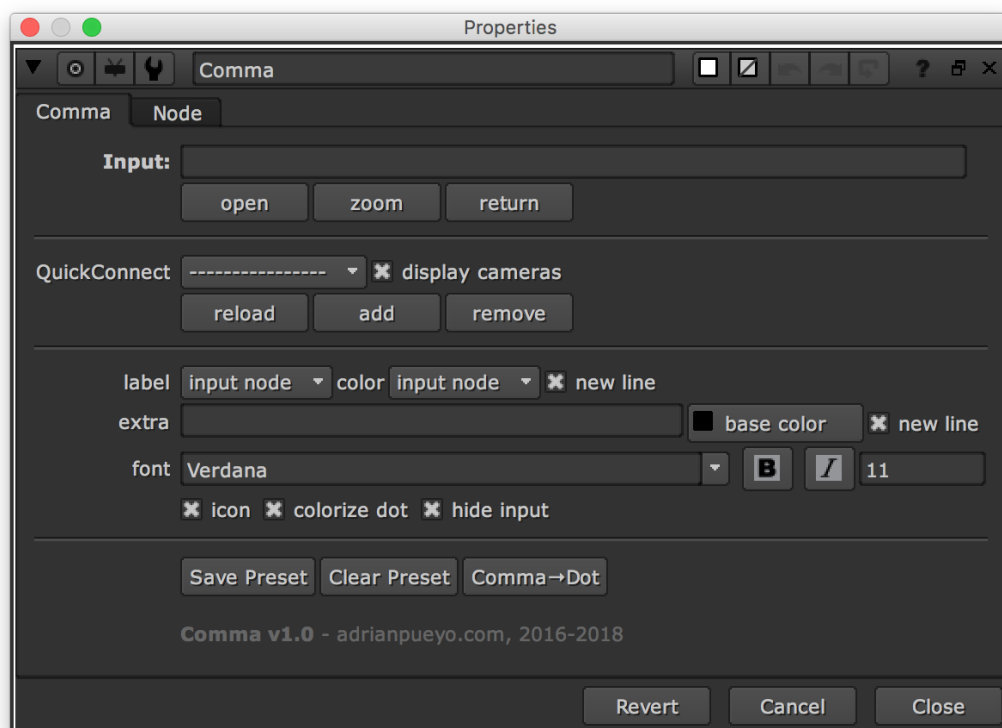
Introduction

Thank you for downloading **Comma**. I hope it'll help you enjoy a slightly faster workflow on your everyday nuke life.

You can also find a video of the tool here: <https://vimeo.com/adrianpueyo/comma>

Comma is a nuke tool that you can think of as an intelligent Dot with its input hidden. Pretty standard, right? For years I used all the similar tools I could find and while working on many different companies and felt they were all lacking some key functionality.

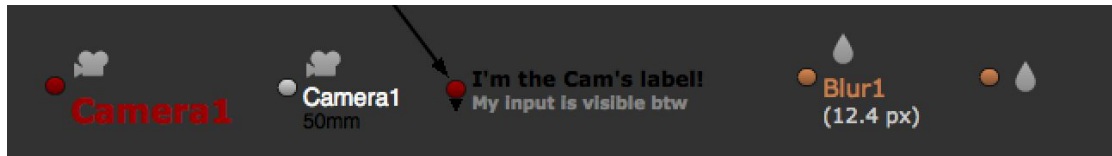
So *Comma* started a few years back as a shitty Dot with hidden input that connected automatically when copy-pasting, which already made it less annoying than most other alternatives, and you called it by pressing the Comma (,) key - and it's been evolving to have more features and be more robust by picking feedback from colleagues and experiences in different places. After wanting to upload it forever, this is the first version that I'm putting out for anyone to try. Should be robust enough. Win/Mac/Linux, Nuke > 9.0.



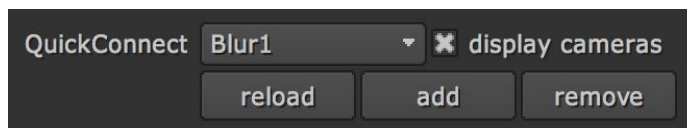
Features

- **Node graph appearance fully customizable.**

Show the input's icon, colorize the dot to the input color, display the input label or name, add extra text or code, etc.



- **QuickConnect list**



...where you can add or remove any node that you'll want to connect new Commas to (i.e. a Camera or a Read

node with your Position or Deep data), and when creating a new Comma you'll see the node in the list for super quick connection.

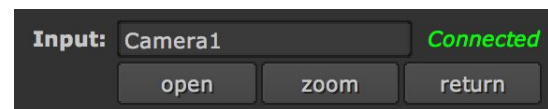
- **Preset functionality.**



The `Save Preset` button stores the current state of the Comma so that whenever you create one it will appear exactly the same - with the same style, and same inputs if you saved it with any. `Clear Preset` will return to the default Comma for newly created ones.

- **Quick access to the input node properties and node graph location.**

Offers a quicker way to check or change settings on the input, or show it on the node graph.



- **Auto-connects when copy/pasting.**

So even if you're copying a part of the node tree with several Commas, you don't have to manually reconnect each one or press any buttons.

- **Fast and lightweight.**

Sort of. Compared to other tools with a similar purpose, and while relying on callbacks, Comma is still fast enough to have been used by me and colleagues in many different places without problems. Happy with render farms too.

- **Can be opened by people that don't have it installed.**

The code each Comma needs to work lives inside of it.

- **Free!**

Would be fun to charge for a Nuke Dot, though.

How to use

A. Creating a Comma

Create a *Comma* by pressing the comma (,) key on your keyboard, or on the nuke Nodes menu: Nodes/Other/Comma (next to the Dot).

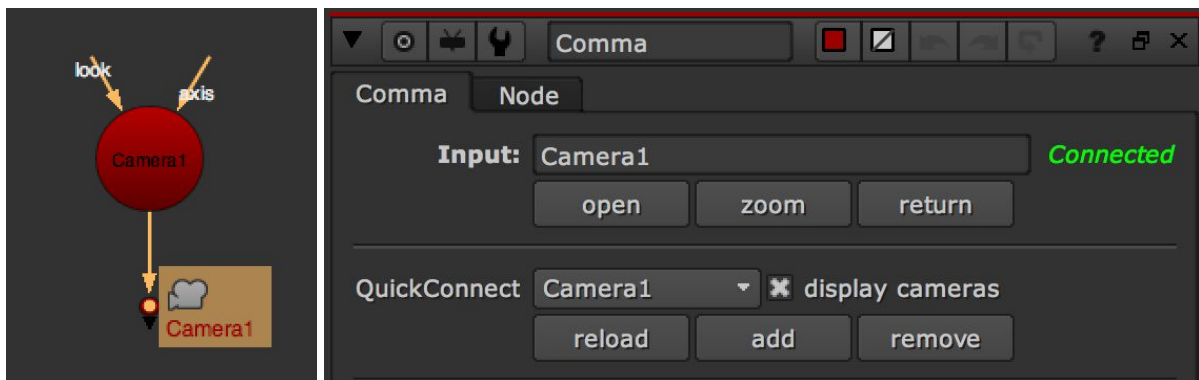
This will make a new Comma on the Node Graph.



B. Connecting to a node

There are four different ways of connecting the *Comma* to a node:

1. By connecting the *Comma* manually like you would do with a Dot.
It will then dynamically update its knobs, store the name of its new input, and change the node graph appearance accordingly.
2. By manually typing the name of the node on the input knob (picture above).
After you press `Enter`, it will get connected and everything else will update too.
3. By using the QuickConnect feature. Simply select the node from the QuickConnect pulldown knob. It will get connected straightaway and update everything accordingly.
4. By copying and pasting a *Comma* that already has an input.



C. Input controls

The `open` button will open the properties panel of the input node.

The `zoom` button will center the input node on the Node Graph.

The `return` button will center the *Comma* on the Node Graph.

D. Using the QuickConnect feature

The QuickConnect list is a convenient way to access some nodes that you plan to use over and over on the script. This is typically the case for a Camera, a Roto, or a piece of geo.

If the `display cameras` knob is enabled, all cameras in the script will appear in the QuickConnect dropdown, even if they aren't on the list. This is normally convenient, except on scripts where there is a large number of cameras (for a reason, I hope). In those cases, I recommend turning the `display cameras` knob off and adding the specific camera you'll use to the QuickConnect list.

- The `add` button adds the current *Comma*'s input node to the Quick Connect list. It will appear there even for newly created *Commas*, as the list is stored in the Root knob.
- The `remove` button does the opposite.
- The `reload` button updates the list in case it has changed.

Tip: If you wish to manually edit the list, open the Project Settings by pressing `s` and navigate to the Comma tab. Then, add the node names separated by commas (funny).

D. Using Presets

Presets are a way to store the current style of a *Comma* you like, and have new *Commas* be created just like that straightaway.

All you have to do is click `Save Preset`.



Note, if the *Comma* has an input when you save the preset, new *Commas* that you create will try to connect to that node. This can be desirable i.e. you'll mostly be using the *Comma* for a single camera on a script. So you make a *Comma*, make it look the way you like, connect it to the Camera, `Save Preset`, and from there all new *Commas* you make will be connected to the Camera.

`Clear Preset` will forget your preset, and new *Commas* you create will be the default one.

`Comma→Dot` is a convenience button to convert the current *Comma* into a normal dot, waaay dumber but also lighter. The dot will keep the same style from the *Comma*.

Installation

A. Fresh install

1. Copy the *Comma* folder and paste it inside your *.nuke* directory.
2. Copy the files *menu.py* and *init.py* inside your *.nuke* folder.
If they already exist, open them with a text editor and add the following line:
On *menu.py*: `import comma`
On *init.py*: `nuke.pluginAddPath('Comma')`
3. Restart nuke.

B. Updating Comma

1. Replace the *Comma* folder inside your *.nuke* directory.
2. Restart nuke.

Update Log

Comma v1.1 (*July 7, 2018*)

- Fixed a display error when copy-pasting with connections.
- Callback fix on copy-pasting without connections.

License



Copyright © 2016-2018, Adrian Pueyo
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

www.adrianpueyo.com