Nuke CTL Plugin User Documentation

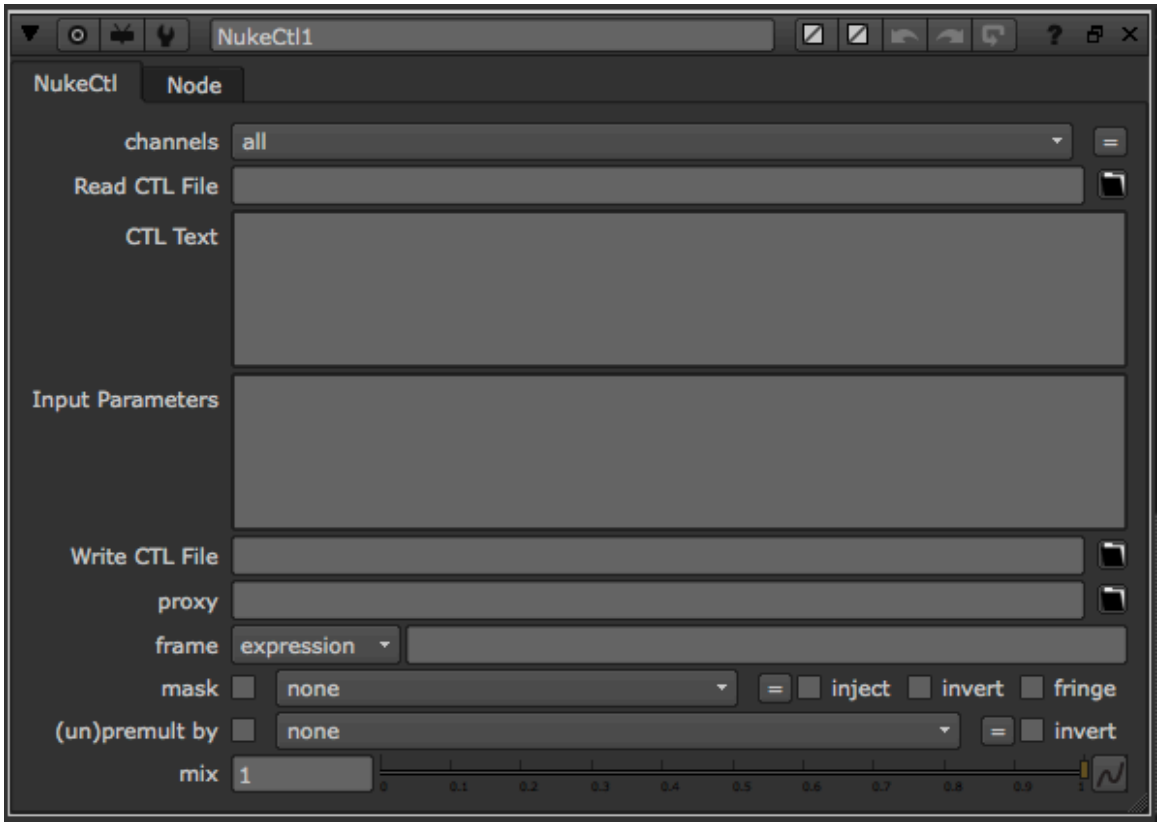
Color Transform Language, or CTL, is a programming language developed as a building block for digital color management. The Nuke CTL plugin allows for CTL use and modifications within the Nuke compositing environment.

1. Loading Plugin

Before using the plugin make sure that the file, NukeCtl.dylib, is installed in the correct directory. This is usually the .nuke folder located in your home directory, but could differ if your NUKE_PATH environment variable is set. Note that the .nuke directory is hidden by default.

Once the plugin is installed in the correct folder, press 'x' within Nuke to open the command prompt. Enter "NukeCtl" to load the plugin into your workspace. The node should be similar to the screenshot below, with four main knobs: Read CTL File, CTL Text, Input Parameters, and Write CTL File.

Create a read node and drag the output arrow onto the CTL node to set the input channels. Drag the output channels onto the viewer or another node to set the output channels.



2. Reading a file

To read a CTL file, click on the folder icon in the “Read CTL File” knob. Select the appropriate CTL file in the file browser and click open. The text data will be loaded into the CTL Text knob. If the file has the appropriate syntax, the transformation will be applied to the image, otherwise an error will be output to the Error Console.

3. Modifying Text

The CTL Text knob displays the text of the current CTL file being processed. This text can also be modified and saved. Once the text has been modified in the knob, click “Write CTL File” and select the filename to save. If the file already exists, the user will be asked if they would like to overwrite the file. To undo or redo a text modification click on Edit->Undo or Edit->Redo.

4. Input Parameters

The “Input Parameters” knob displays input parameters specified by the user. By default rIn, gIn, bIn, rOut, gOut, bOut, and aOut must be present, but other variables may be added to be processed in the CTL functions. These variables will be automatically detected by the plugin and will appear in this knob. The user can then modify the default value of 0, and see the updated results. If an input parameter is an array then a 0 will be output for each element of the array.

Note: If an input parameter is added for modified, reload the file following step 2. When reading a file, the plugin allocates memory for each input parameter. If the parameters in the CTL text are modified and not reloaded, Nuke may try to access unreserved memory and crash.

5. Masking

Masking can be applied by creating a roto node and piping its output into the mask channel of the NukeCtl node.

6. Known Issues

Currently the plugin may crash Nuke by switching the “Channels” knob attribute. If an issue is found please report to https://github.com/ampas/CTL/issues