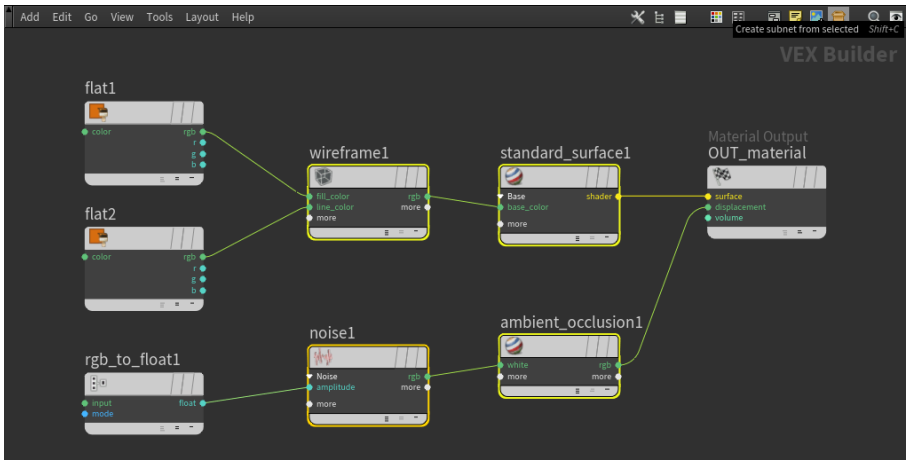
VOP Shader Compounds

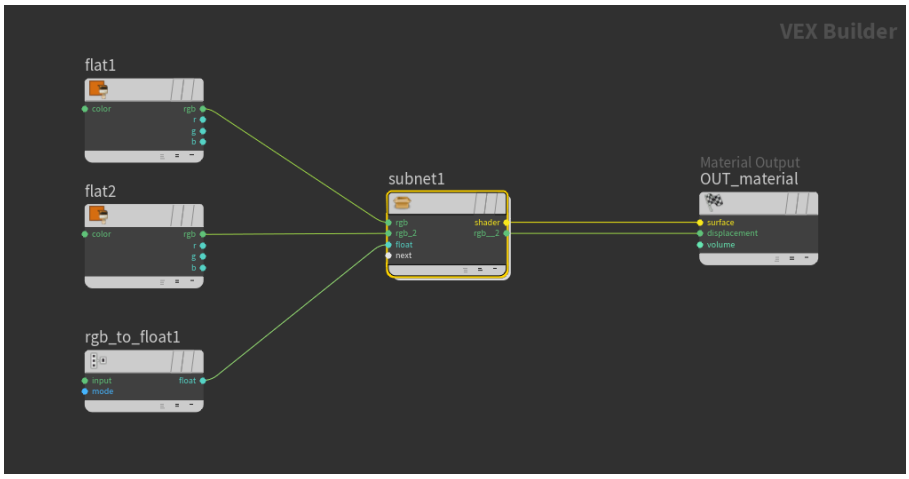
Creating a Subnetwork

To create an HDA, all the parts of your digital asset must be grouped in a subnetwork operator.

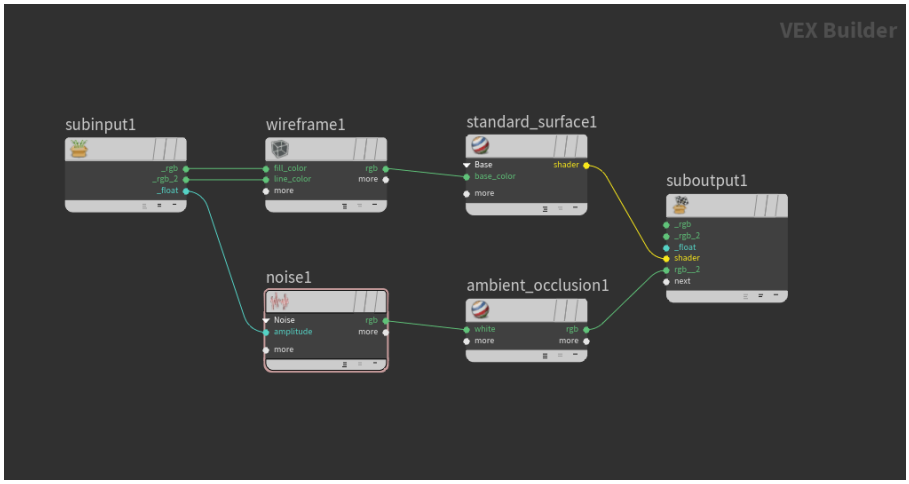
A subnetwork can either be created from scratch or from a pre-existing shader network using the **create subnet from selected** button. You will need to know which parameters are going to be exposed. To parameterize selected inputs add connections that won't be included in the subnet selection.



Selecting the nodes and clicking the subnet icon



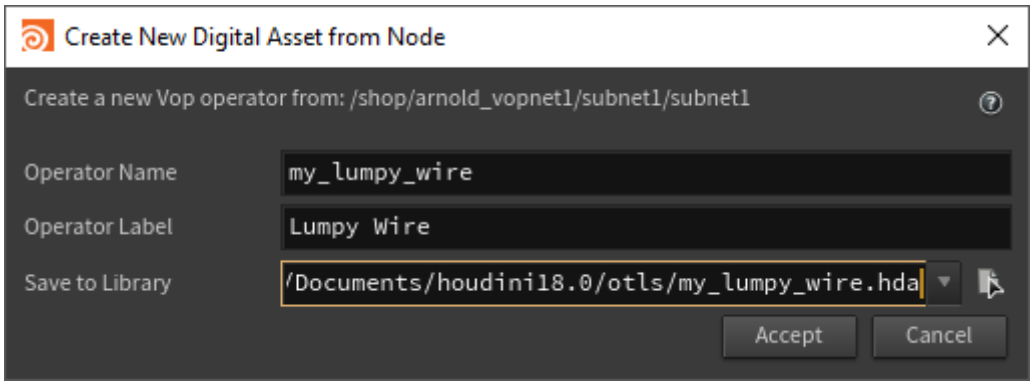
A subnet has been created



The nodes inside the subnet

Creating an HDA

From the subnet, a digital asset can be created. Right-click on the subnet and choose **Create Digital Asset** which brings up the **Create New Asset from Node** dialogue.

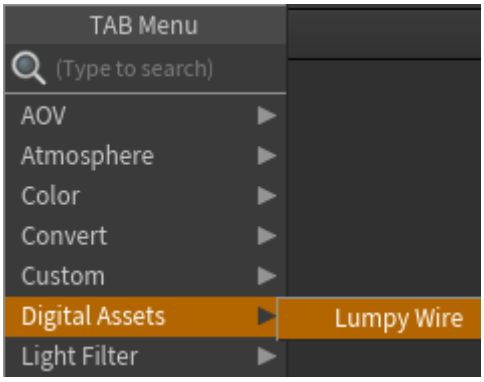


The HDA must be given a unique Operator Name. It must not have the same name as any other operator in Houdini as it will override it. For safety, it is a good idea to prefix the HDA with your username, show or shot depending on its use.

An OTL is a library of HDAs. The HDA can be added to a pre-existing library although unless it is going to be part of a sophisticated pipeline that requires otherwise it is recommended to save one HDA per OTL with the OTL given the same name as the HDA. The OTL will be saved into /home /user/houdini/otls unless specified otherwise.

Clicking **Accept** will save the OTL file and bring up the **Edit Operator Type Properties** box. This will be discussed more in the next section.

The HDA will now be available in the Tab menu.



More information can be found in the Houdini documentation: Creating a digital asset.

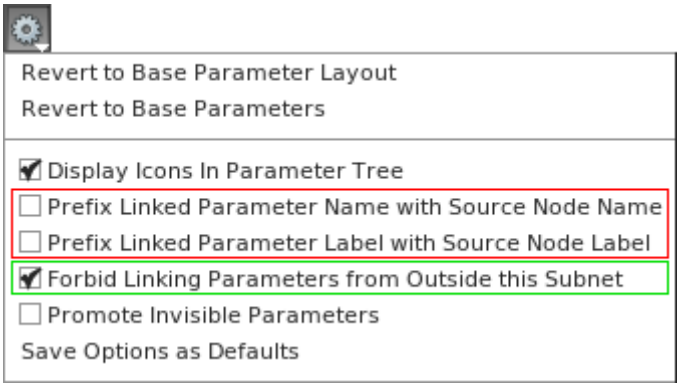
Editing HDAs

While working on an asset you will be able to lock, unlock and edit the digital asset by entering the subnet to alter nodes. A blue title indicates that the HDA is locked and cannot be changed whilst a red title shows it is editable and the internal network may differ from the original HDA. Once the changes have been made to the HDA; right-click the node > **Save Operator Type**. To update duplicate HDAs in the scene to be the same as the saved version; right-click > **Allow Editing of Contents**; right-click > **Match Current Definition**.

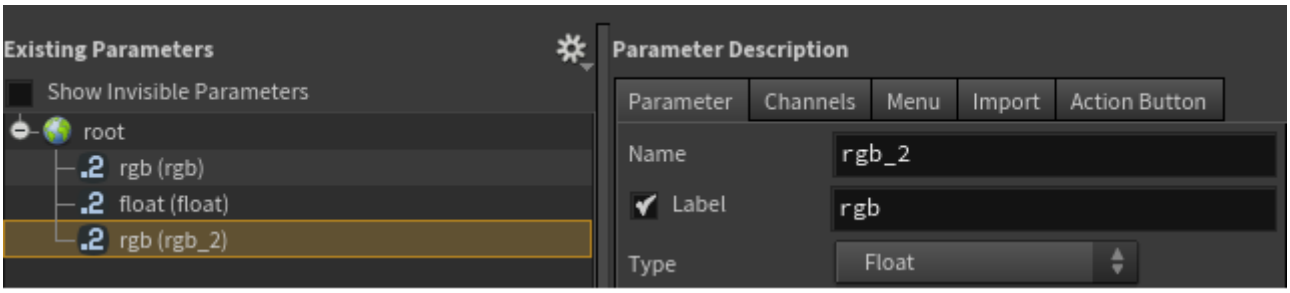
HDA Properties

The inputs for the HDA will show the data type but it would more useful to give it a name to what it is connected. To edit a digital asset; right-click > **Type Properties**; and click on the **Parameters** tab.

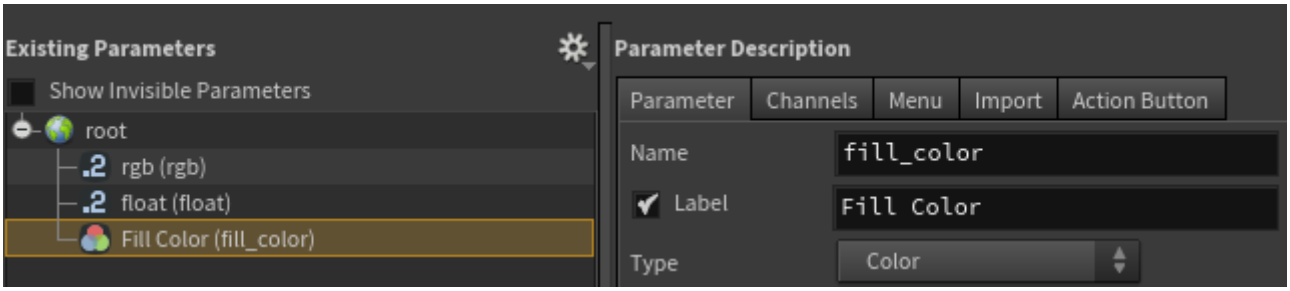
Before adding any parameters it's advisable to make sure the **Prefix Linked...** options are unchecked in the options and **Forbid Linking...** is turned on.



Choosing a parameter in the **Existing Parameters** column allows the names to be changed. *Name* is the internal Houdini token whilst *Label* is the nice name for the interface. The parameter *Type* can be changed, for example from *Float* to *Color*. It is also possible to add range limits and interface options.



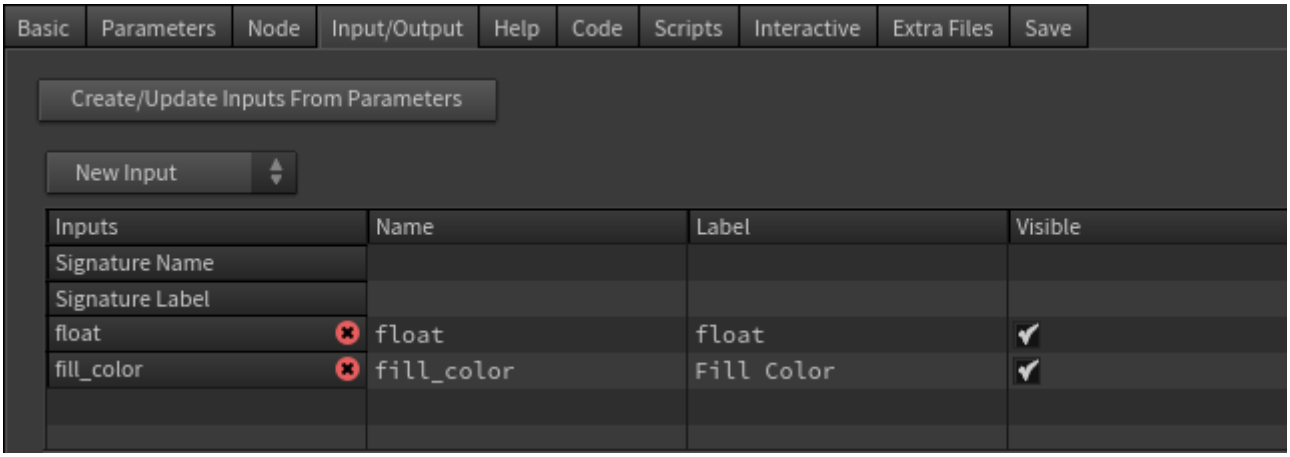
Default parameter names and types



Parameter names and types changed

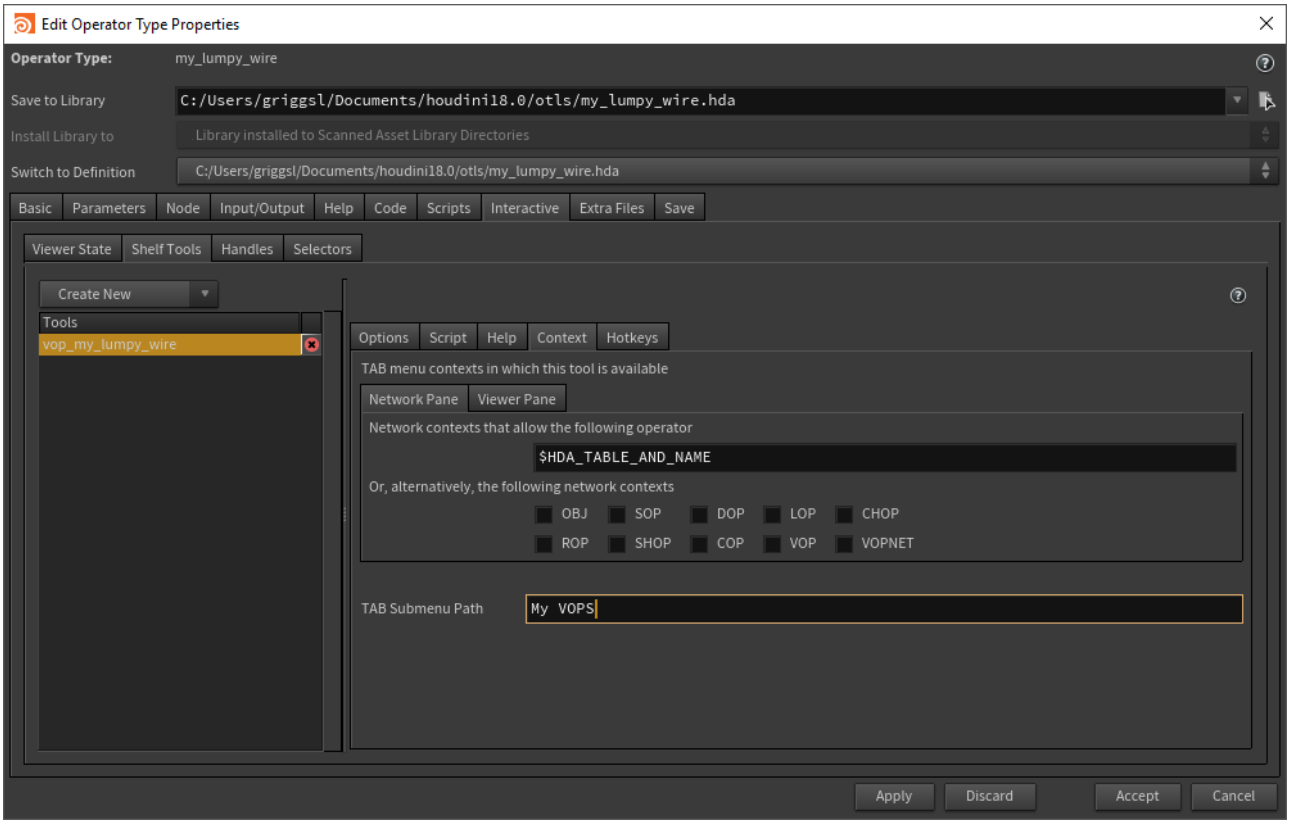
Matching Inputs

It is important that the input must have the same name as the corresponding parameter. Choose "**Create/Update Inputs From Parameters**" from the **Input/Output** tab. This will add the updated names as *Inputs*. Delete the old inputs using the cross.



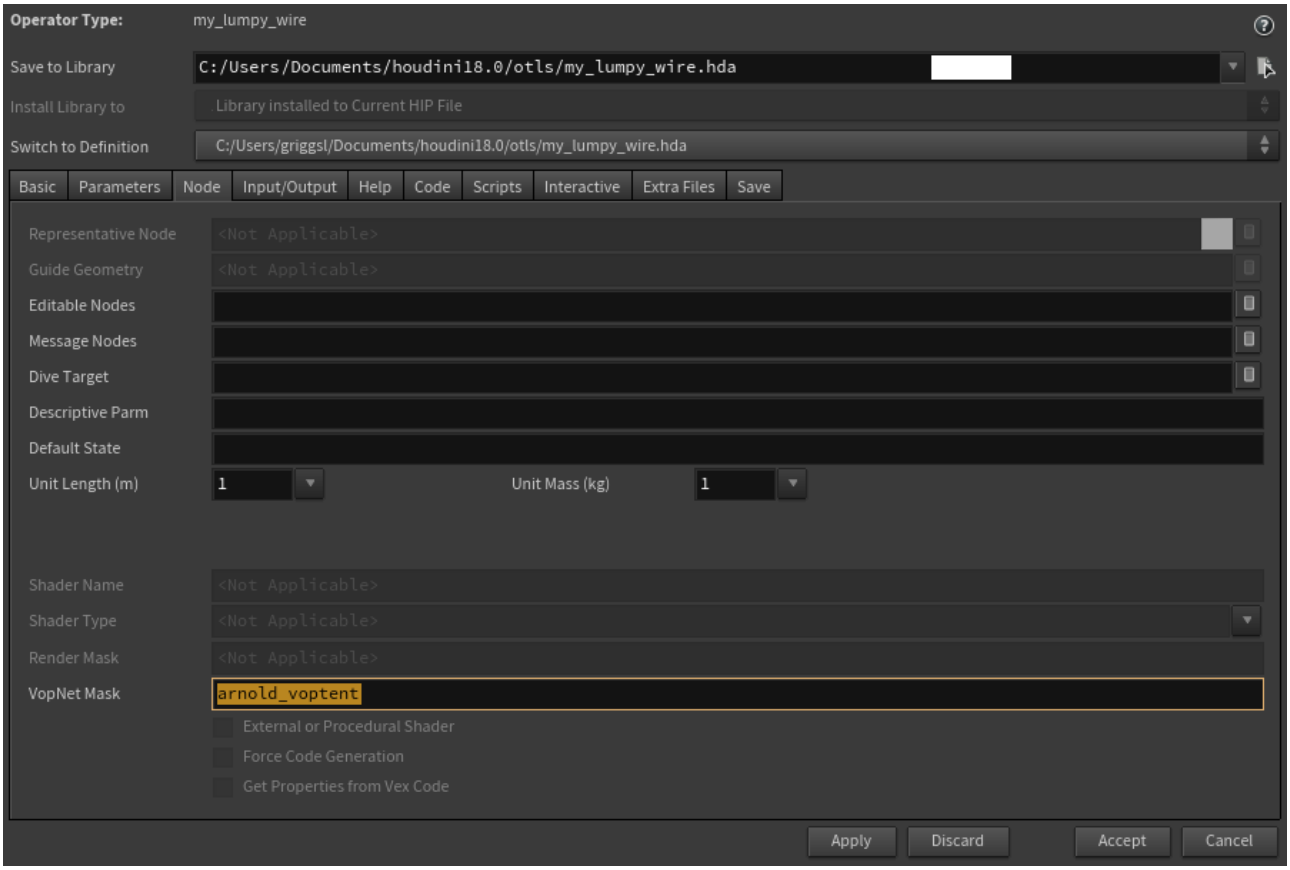
Changing Tab Submenu

The Tab submenu can be changed in the ***Shelf Tools Tab > Context Tab***; and then change "**Digital Assets**" in the ***Tab Submenu Path***.



Setting The Vopnet Mask

Under the **Node** tab, the ***vopnet_mask*** sets where the HDA will be available. If blank it will appear everywhere including Houdini SOPs. Enter **arnold_vopnet** to constrain it to Arnold shader networks.


Operator Type:
my_lumpy_wire
Save to Library
C:/Users/Documents/houdini18.0/otls/my_lumpy_wire.hda
Install Library to
Library installed to Current HIP File
Switch to Definition
C:/Users/griggsl/Documents/houdini18.0/otls/my_lumpy_wire.hda
Basic
Parameters
Node
Input/Output
Help
Code
Scripts
Interactive
Extra Files
Save
Representative Node
<Not Applicable>
Guide Geometry
<Not Applicable>
Editable Nodes
Message Nodes
Dive Target
Descriptive Parm
Default State
Unit Length (m)
1
Unit Mass (kg)
1
Shader Name
<Not Applicable>
Shader Type
<Not Applicable>
Render Mask
<Not Applicable>
VopNet Mask
arnold_voptent
External or Procedural Shader
Force Code Generation
Get Properties from Vex Code
Apply
Discard
Accept
Cancel