

Vex/Vops

by Deborah R. Fowler

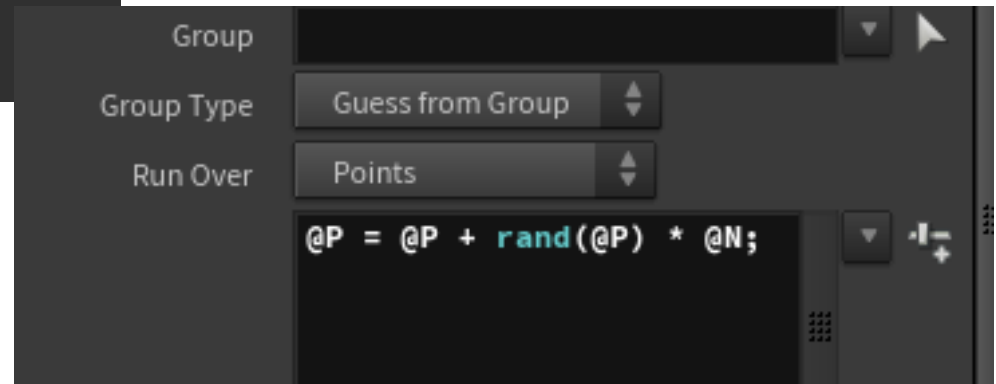
move a point along a normal

- first pointwrangle
- next in vops
- next in vex
- in python

Point Wrangles



Currently (H16) this is the preferred method



Side note about point wrangle nodes

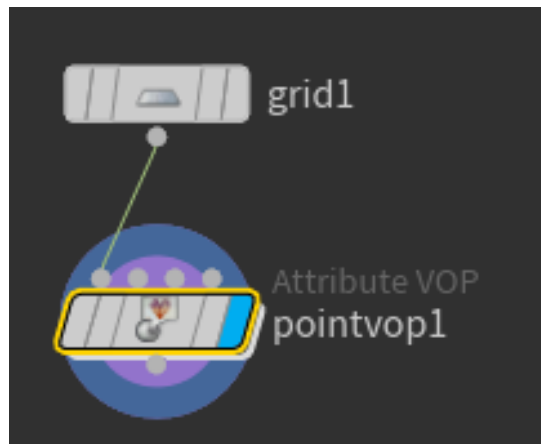
Could also use a point node, but the pointwrangle node is faster by about 10x. Once you are used to the syntax, use pointwrangle in favor of point. The syntax of the @ sign is an attribute and can be used to create or fetch (in the code below it is setting the normal to be random based on the point)

```
@N = rand(@P);
```

Note in H13 only @N = rand(P); works but not in H14/15

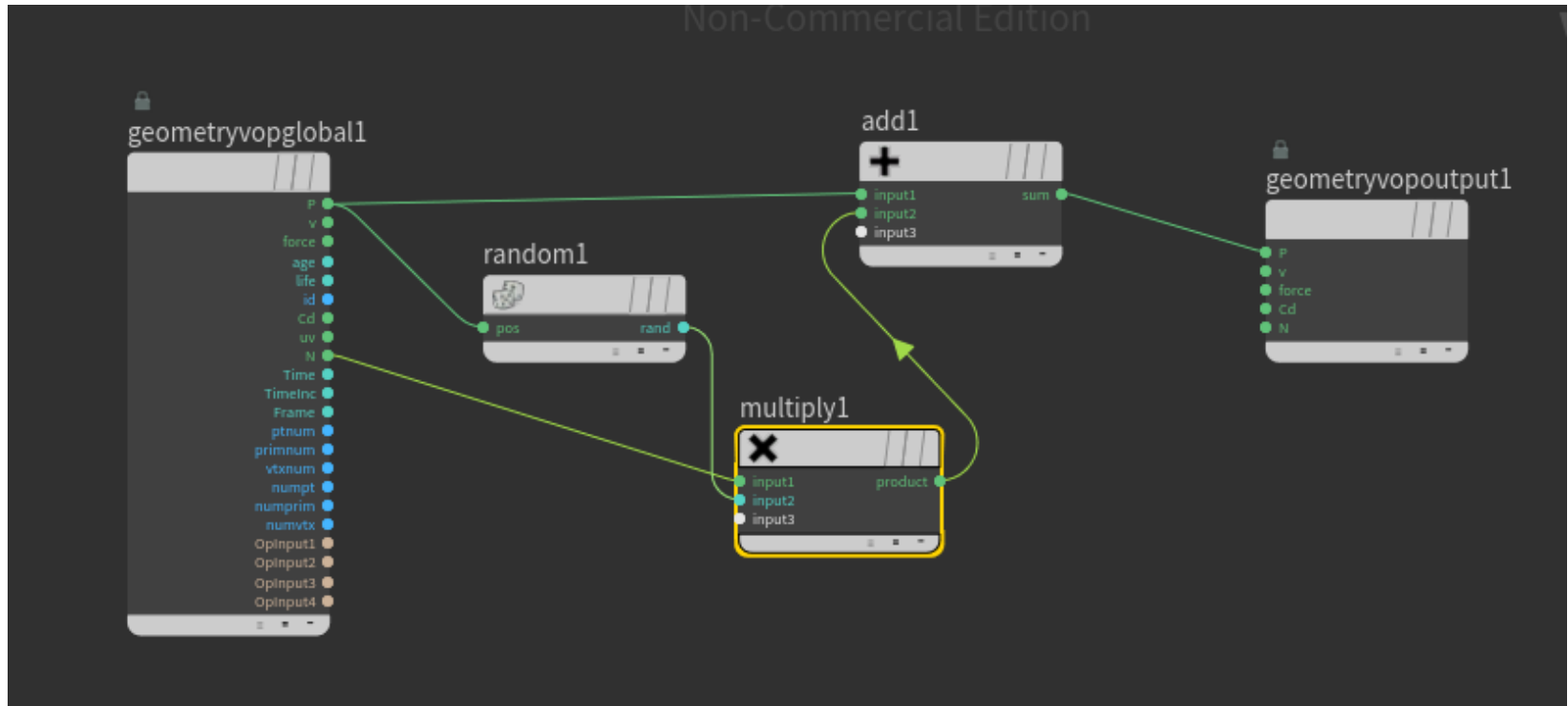
Point Vop

- hit tab and type in point vop (H13 it was vop sop)



- dive into the point vop and you will see geometryvopglobal1 and geometryvopoutput1 – we are going to add the N to P and pipe that back into P

in a point vop $P = P + \text{rand}(P) * N$ looks like this:



Note that the colors are meaningful, green is a vector, teal green a float and blue an integer value and pale orange a string (dashed lines will indicate there is a type conflict)

now you have something that sort of acts like a
mountain node

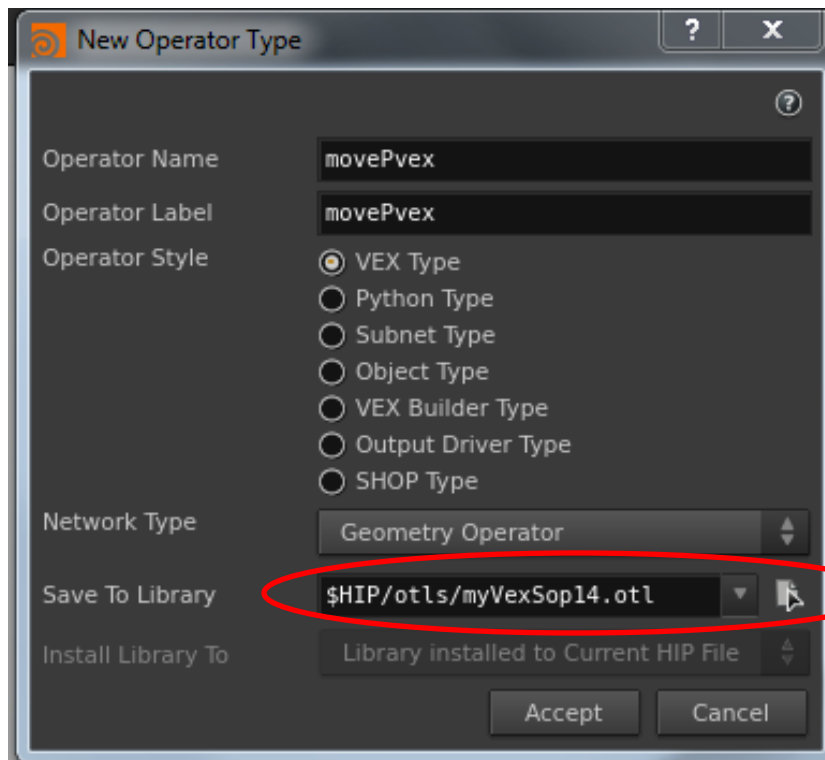
Let's do the same thing in vex

- vops are the pictorial representation of vex
- vops nodes represent code

The vops are compiled to vex code (vcc is a script that is used for this that exists in Houdini bin)

Create a vex node prior to H16 (see next slide for H16)

File/New Operator Type



Make sure you rename your otl (hda) and put it somewhere relative to your hip so you can find it

The Geometry Operator type no longer exists in H16 <https://www.sidefx.com/forum/topic/50606/>

Easiest way to do this now is to create a basic hda

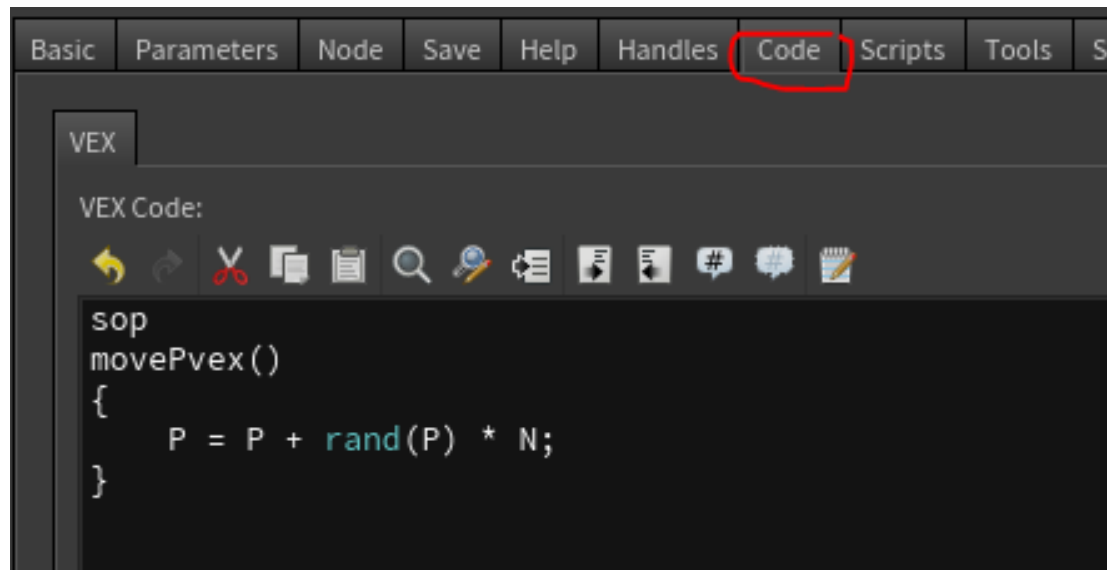
```
sop mysop()  
{  
}
```

- Put this into a vfl file.
- Run **vcc -l mysop.hda mysop.vfl**

Then install the digital asset and proceed as before.

in the vex node

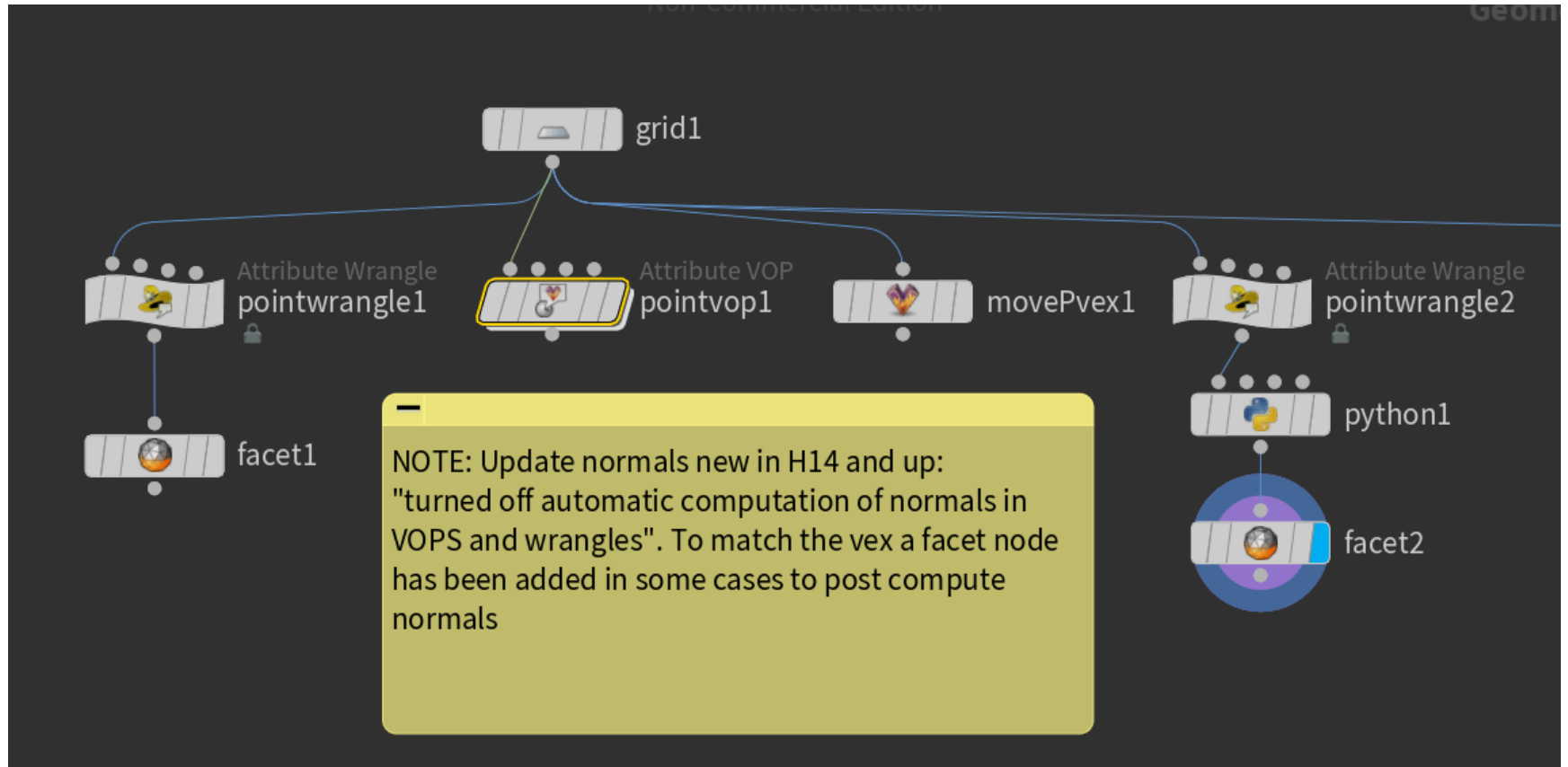
- click on Edit Vex Function
- now type into the function “sop” under the code tab



Important

- vex is stored in the otl (or hda), vops are stored in the hipnc and converted to vex automatically
- hda (previously labeled otl and still stored in otl directories) are very handy for sharing assets

Note about Normals



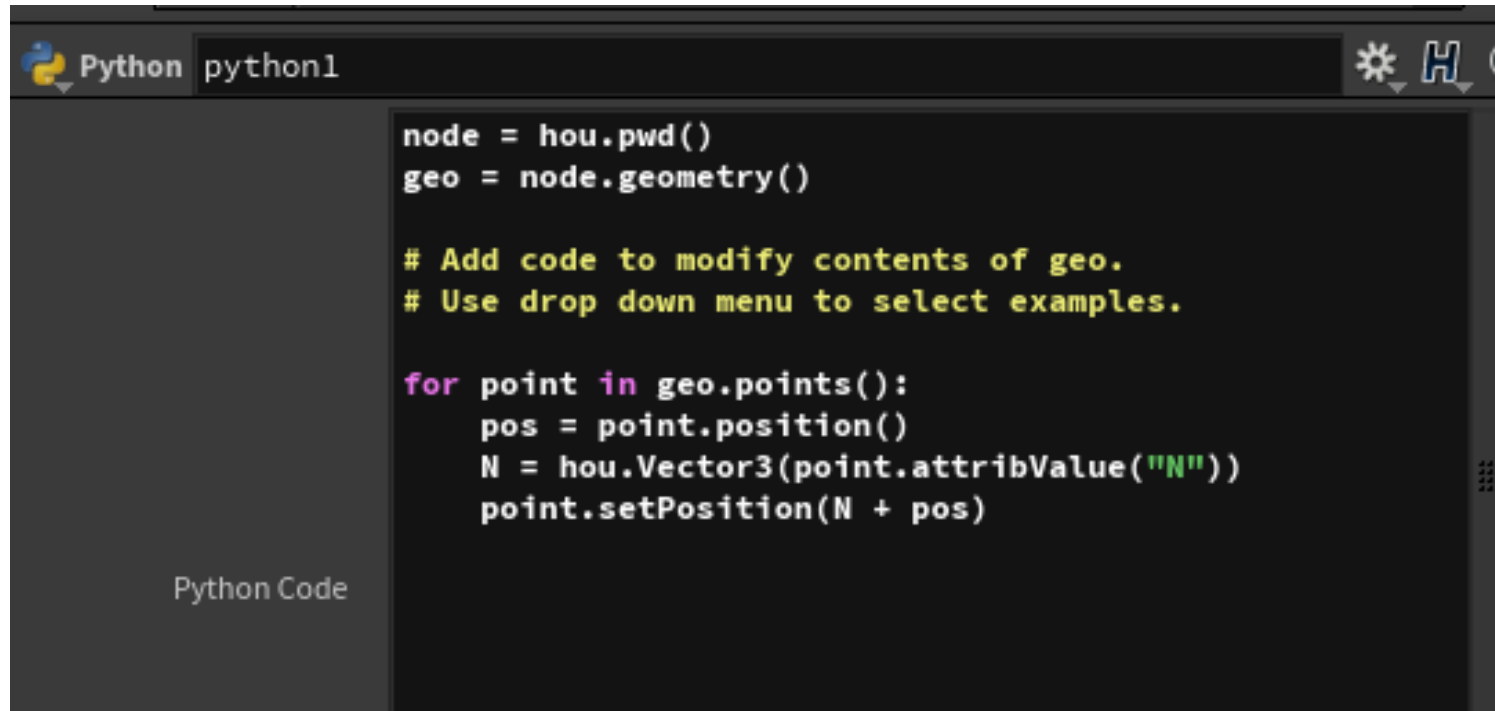
Now in Python

- hit tab and type python
- or File/New Operator Type and select python (similar to the vex)

Two things to note:

- python uses calls to HOM (Houdini object module)
- python and vops do not compute the normal after so we need to add a facet or fuse to do this (see hipnc) (in H13 vex/vops both recomputed normals, now only vex does)

python version



```
node = hou.pwd()
geo = node.geometry()

# Add code to modify contents of geo.
# Use drop down menu to select examples.

for point in geo.points():
    pos = point.position()
    N = hou.Vector3(point.attribValue("N"))
    point.setPosition(N + pos)
```

Python Code

Note that vex and vops operate on each point, in python we explicitly loop thru the points, grab the position and normal information in the correct form and set the point position to the new value. Here we used a point wrangle node before to set the random normal. (see hipnc for more details)