

Houdini

Tips for the Beginner

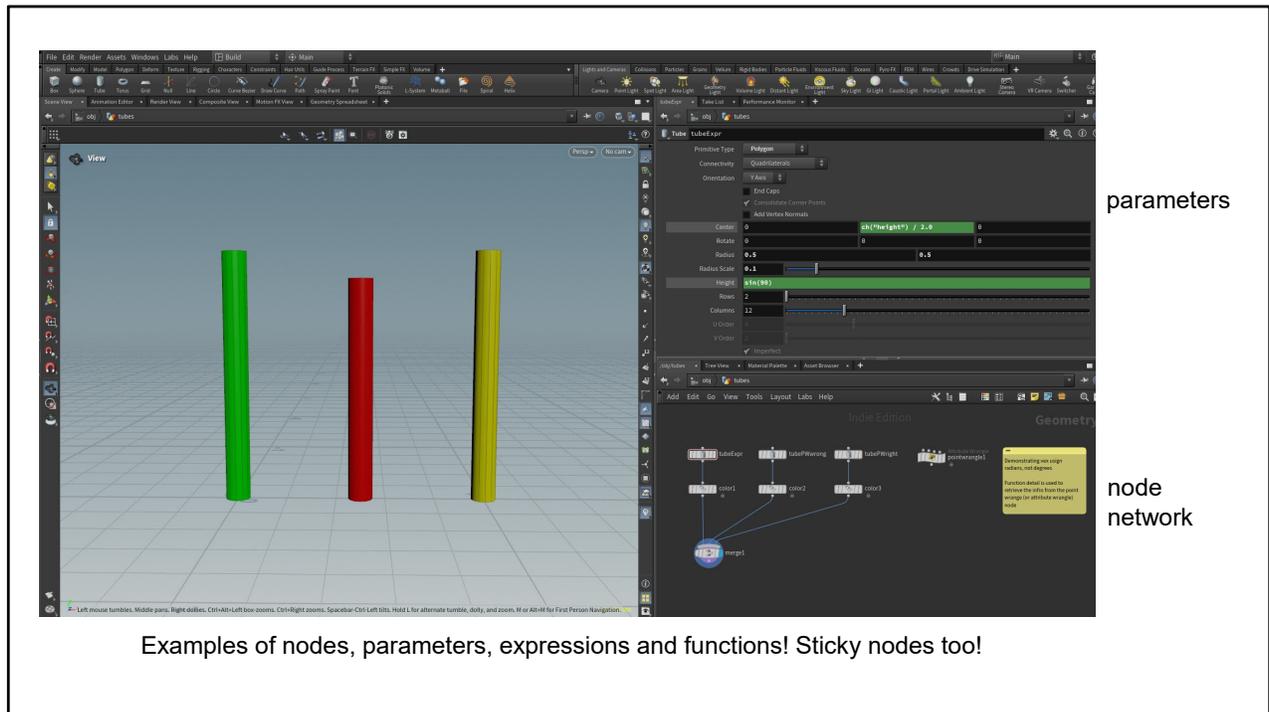
- easy to learn
- node based
- attributes
- contextual
- functions do things for you
- multilingual
- modular
- lots of nodes
- custom nodes

easy to learn

Ignore the hype – yes, Houdini has many aspects, but you don't learn it all at once – it is beautifully modular and powerful – the following cover concepts to keep in mind as you journey into the software

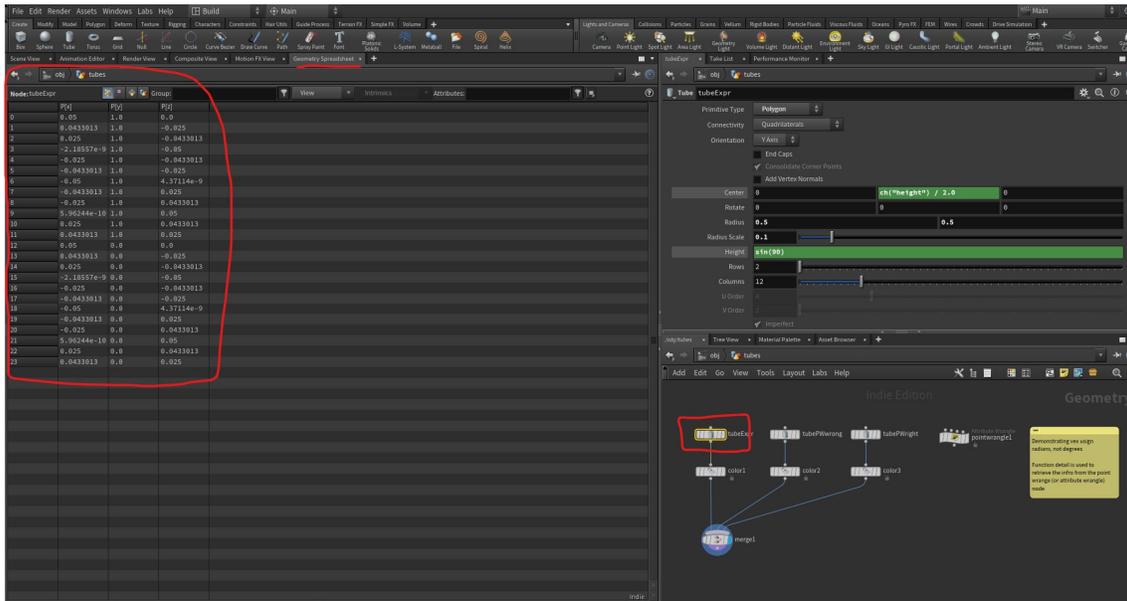
node based

It is node-based – like nuke and unreal blueprints, and even Maya (according to its documentation, although really only appears evident in hypergraph)



Here we have nodes that represent tubes with expressions in the `ty` parameter, and height parameter. The expression `sin(90)` uses the `sin` function. The expression `ch("height")/2.0` uses the `ch` function to retrieve the value of height.

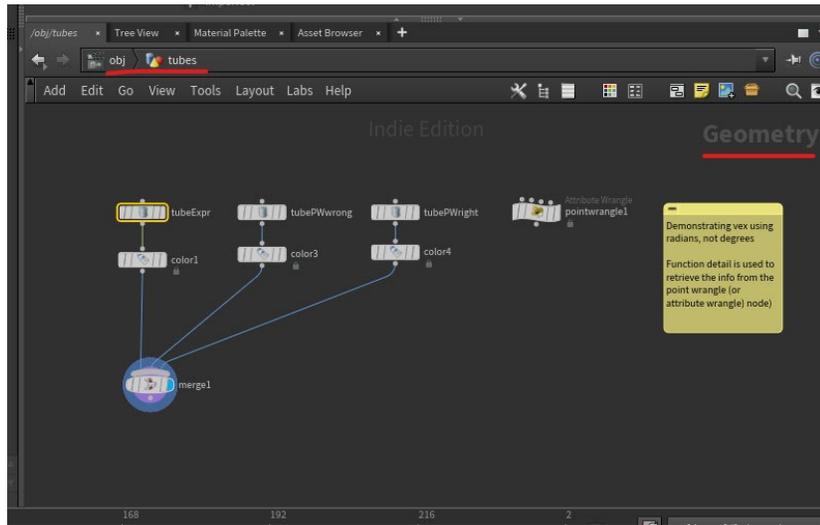
attributes



Geometry spreadsheet display point (shown) as well as vertex, primitive and detail
Think of attributes as information that follow the node around

Attributes are information stored with the node, so for example when you uv you are adding a uv attribute, if you add a color a color attribute, and so on. The geometry spreadsheet can be helpful for understanding this. There are types of attributes: point, vertex, primitive and detail. This relates to our next concept – context.

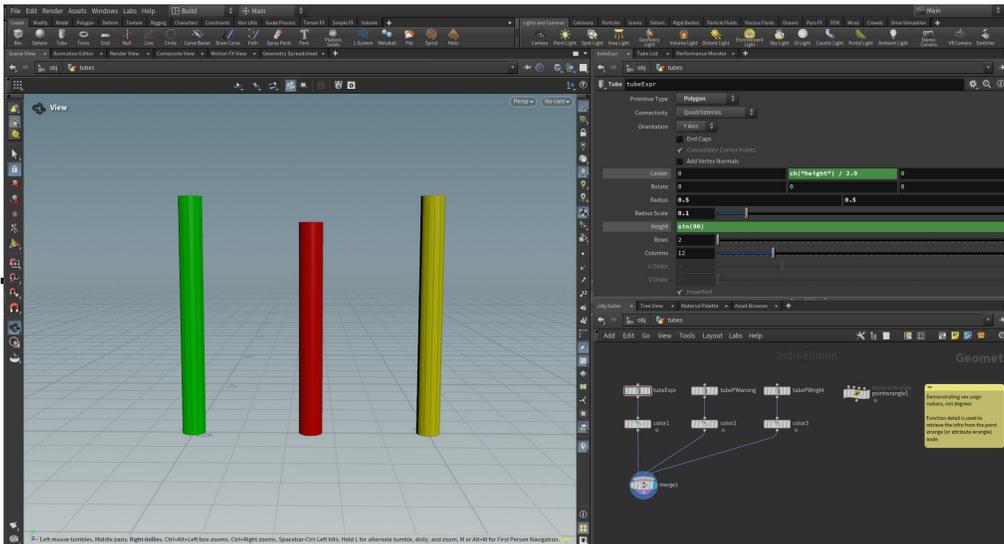
contextual



Most commonly you are in the obj/geometry context – but you can be in other contexts as well such as “out” for rendering, “mat” for shaders, etc.

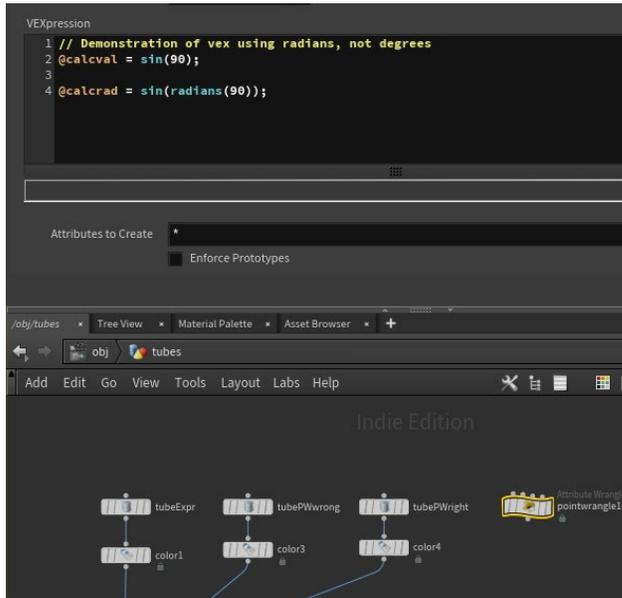
We will mostly be working in the geometry context. The top level is obj where everything in your scene is and as we explore you will be introduced to other contexts like dynamics, rendering and so on.

function do things for you



Functions are a concept from programming class that will help you understand expressions. Inline functions go off and return something. For example, if you give sin and argument $\sin(90)$ it will return a value (1 if it is expecting degrees and .893997 if it is expecting radians).

multilingual



In Houdini you can use multiple languages. There are many ways to add functionality and proceduralism. We will start with hscript and vex, but you can also use python, vops and so on.

Shown is an example of vex code which is adding a detail attribute that can be referenced elsewhere.

modular

The reason Houdini is easy to learn is it lends itself very well to modular design – breaking down a big problem into smaller manageable problems.

lots of nodes

Yes, there are a lot of nodes. You will continue to discover the more you use Houdini - you will learn the commonly used nodes, but if you are in need of something - take a look at the tab menu - likely someone has provided that functionality for you.

Hit Tab and type the letter “c”

custom nodes

If not, as you become more advanced users you can also easily build your own. We'll discuss SideFX labs and HDAs.