Subdivision Surfaces in Katana

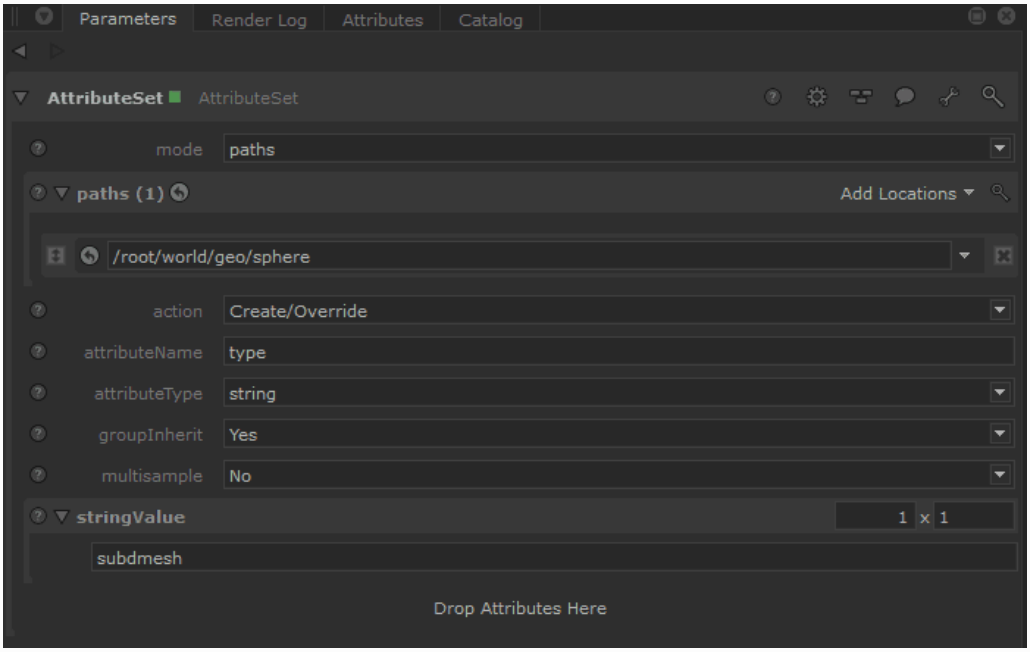
Subdivision surfaces provide you with a great way to use the simplicity of polygonal-style modeling, together with the smoothness of a mathematically defined surface. They are used heavily when modeling organic or curved surfaces. The RenderMan documentation for Subdivision Surfaces will give you a great primer on them.

In Katana, you will most likely bring in a subdivision surface via Alembic or some other geometry format that supports them. You as an artist are not likely to have to do anything special to utilize them in Katana. In Maya, using the RenderMan Archive workflow, you can embed the settings used on your geometry and RfK should simply use them as set.

Any texture coordinates or other primitive variables that you need for look development must be added upstream in your pipeline. They will be available when you import your geometry into Katana.

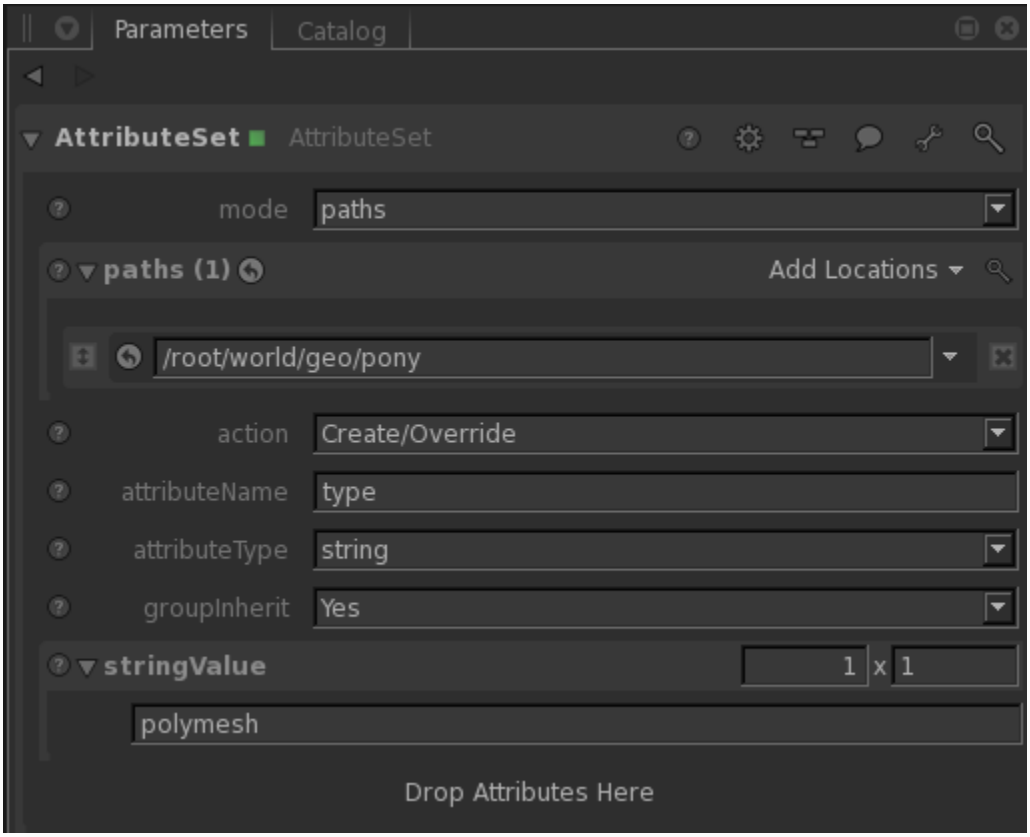
To set an object that is ingested as a polymesh into a subdivision mesh requires you use an AttributeSet node with the string value set to subdmesh and the path to your required object. The PrmanObjectStatements can be added to specify the type of SubD Mesh (Catmull-Clark, Loop, etc.) The default is Catmull-Clark.

However, the OpScript provided below on this page should match the default from Maya and Houdini should you need to match them (recommended)



Speed Trick: If your subdivision surface is off-screen or the pixel coverage is very small, it may make sense to convert the subdivision surface to a polygonal surface. This will save memory in your render, will shorten the time to seeing the first pixel, and depending on your scene complexity, will reduce render times.

To change your geometry from a subdivision surface to a polygonal surface, use an AttributeSet node to set the "type" of the geometry to polymesh. Below is a screen snapshot of what you will need to do.



Normals and faceting

When rendering surfaces in Katana, you may need to discard normals. The basic workflow for rendering smooth-looking surfaces is below:

- Export normals for polymeshes with smoothed normals that you wish to preserve when they are *not* going to being subdivided. This is a common "fake" for low-poly surfaces that need to look smooth. No normals export is needed when you only require hard edges/faceting.
- Going to subdivide the mesh? Do not export normals for these polymeshes, let PRMan calculate them new when they are subdivided. This also saves space for the exported file since you don't need the normals.

The presence/absence of normals may affect tessellation. Otherwise, remove unwanted faceting by deleting normals in Katana if they were exported despite your intent to subdivide.

Displacement

To match Displacement from previous versions of RenderMan, the following conversion may be of use:

To get a similar tessellation level, using a shading rate (deprecated) value of X would translate in using a micropolygonlength value of sqrt(X):

shading rate = 0.25, micropolygonlength = 0.5
shading rate = 1, micropolygonlength = 1
shading rate = 4, micropolygonlength = 2

To Duplicate the Geometry Settings from Maya

It may be useful to match what you're seeing in Maya since it may be your primary modeling package. Below are the settings necessary to achieve the same look if you asset was not exported with the data.

1. Create an OpScript node
    a. In the CEL section, select the mesh(es) where you wish to apply the attributes. In this case we've added the subdmeshes from Maya
        i. Custom CEL statements
            1. /root/world/geo//*{@type == "polymesh"}

2. /root/world/geo//*{@type == "subdmesh"}

ii. In the script section add the following:

```
Interface.SetAttr('geometry.facevaryinginterpolateboundary', IntAttribute(3))
Interface.SetAttr('geometry.facevaryingpropagatecorners', IntAttribute(0))
Interface.SetAttr('geometry.interpolateBoundary', IntAttribute(1))
Interface.DeleteAttr('geometry.vertex.N')
Interface.SetAttr('type', StringAttribute('subdmesh'))
```

2. We set several attributes above to match the Maya settings. Below we list the other options should you need to experiment.
    a. Options for geometry.facevaryinterpolateboundary
        i. 0 (old style)
        ii. 1 (new style)
        iii. 2 (new style, no corners)
        iv. 3 (new style, smooth internal only) *This is the current Maya default*
    b. Options for 'geometry.facevaryingpropogatecorners'
        i. 0 (off) *This is Maya default*
        ii. 1 (on)
    c. Options for 'geometry.interpolateBoundary'
        i. 0(no interpolation)
        ii. 1(edge crease, corner crease) *This is the Maya default*
        iii. 2(edge crease only)
    d. geometry.vertex.N
        i. We delete the existing normals off the mesh as RenderMan will create its own for a subdivision mesh. This is explained in the above section.
    e. type
        i. Changes the mesh to a subdivision surface

For Developers

If you are implementing support in a custom Katana plugin for subdivision surfaces, you will be interested in how RenderMan for Katana translates geometry attributes to RenderMan's subdivision format. See the developer documentation here.

We support the following attributes, which are converted to RenderMan's internal format.

- prmanStatements.primAttributes.subdivisionMesh.scheme

- geometry.holePolyIndices

- geometry.creaseLengths
- geometry.creaseIndices
- geometry.creaseSharpness
- geometry.creaseSharpnessLengths

- geometry.cornerIndices
- geometry.cornerSharpness

- geometry.facevaryinginterpolateboundary
- geometry.facevaryingpropagatecorners
- geometry.interpolateBoundary