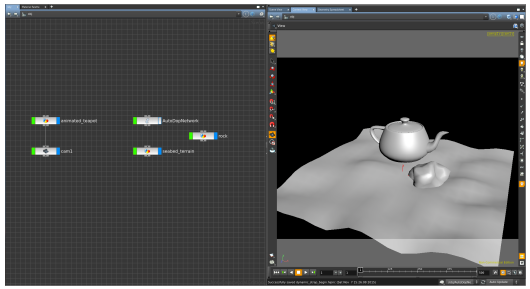
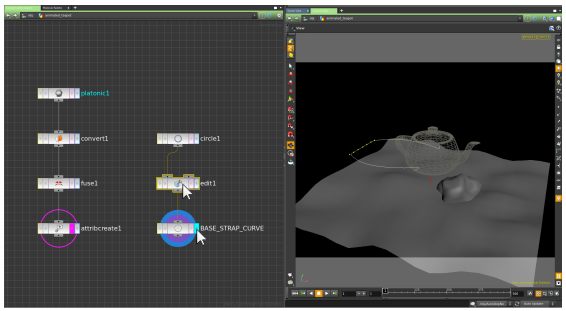
CREATING A DYNAMIC STRAP

Open the scene dynamic_strap_begin.hipnc. This scene contains an **animated_teapot** object going back and forth over a **seabed_terrain object** and a **rock object**. The **animated_teapot** and **rock** objects have been already configured as **dynamic Static Objects**, and the **seabead_terrain** has been configured as a **dynamic Terrain Object**. The **nodes** for this **dynamics setup** can be found inside the **AutoDopNetwork**.



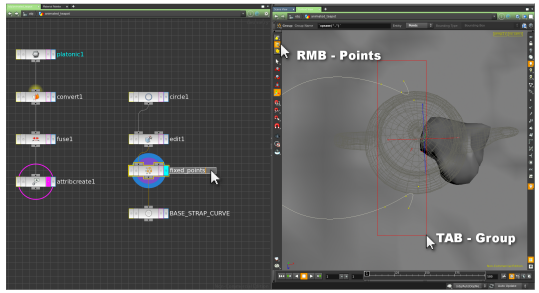
This example will look at the creation of a dynamic strap that will attached to the teapot, causing it to interact with the other dynamic objects as the teapot animates. As with the **dynamic chain example**, a **NURBS base curve for the dynamic strap** has been **configured inside** the **animated_teapot object**.



PREPARING THE CURVE

As the strap will be **dynamically attached** to the teapot, **the attachment points** of the **base curve** need to be **grouped**. Set the **Display / Render Flag** to the **edit1 node**, and press **ENTER** with the **mouse over the Viewer** to activate the **Edit Tool Mode**.

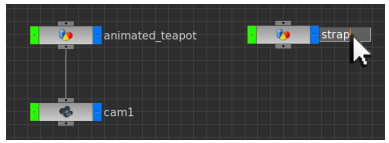
Activate a **Point Geometry Select** Mode, and whilst looking down on the base curve, **select** the **first four** and **last four points** of the base curve.



With the **points selected**, press **TAB** and **type group** with the **mouse over the Viewer**. Rename the resulting **Group SOP node** to **fixed_points**.

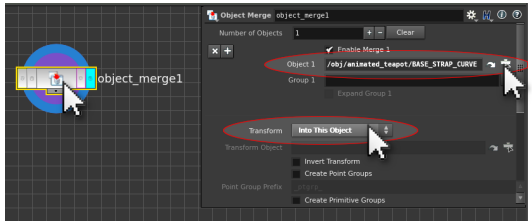
NOTE: This renaming of the **Group SOP** only works if **`opname(".")`** has been assigned as the **Group Name** parameter as it's **Permanent Defaults**.

Reset the **Display / Render Flag** back to the **attributecreate1 node** of the main teapot network, and at **Object Level** create a **new geometry object** called **strap**.

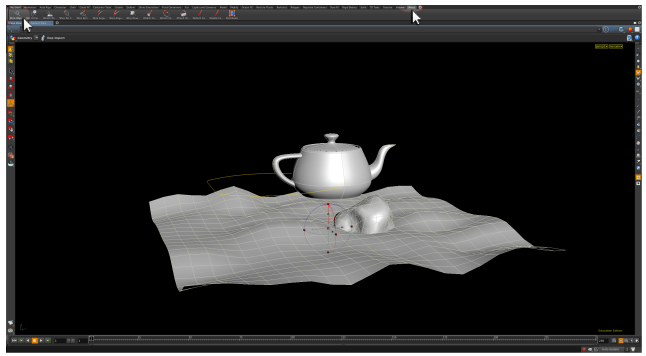


Inside it, **delete** the default **File SOP** and create an **Object Merge SOP**. In the **parameters** for the **Object Merge SOP** specify:

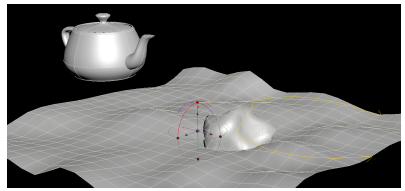
Object 1	**/obj/animated_teapot/BASE_STRAP_CURVE**
Transform	**Into This Object**



This will read in the base curve into this new object, as well as the animation assigned to it by the animated_teapot. At **Object Level**, **select** the **strap object node** and **maximize** the **Viewer**. From the **Wires Shelf**, **activate** the **Wire Object button**. This will **import** the strap object into the **AutoDopNetwork**, configuring it as a dynamic wire that can interact with other dynamic objects.



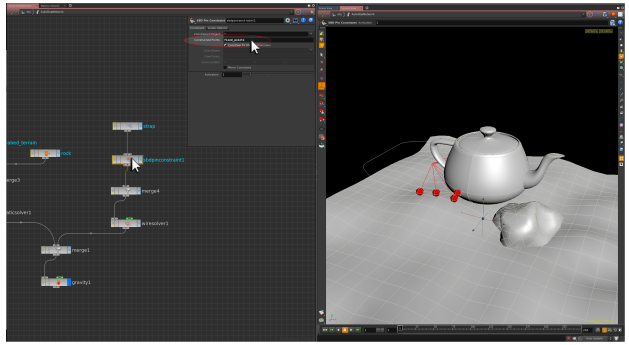
When **PLAY** is pressed, the dynamic wire attempts to remain attached to the teapot (due to a Gravity DOP value of -3), but eventually separates and falls away.



FIXING THE DYNAMIC CURVE TO THE TEAPOT

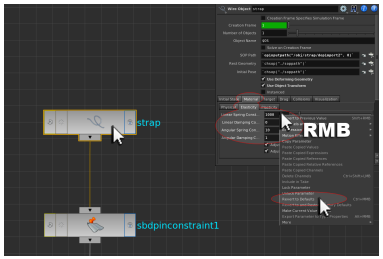
Rewind the **simulation**, and **go inside** the **AutoDopNetwork**. **Append** to the **strap Wire Object DOP** a **SBD Pin Constraint DOP**. In the **parameters** for it specify:

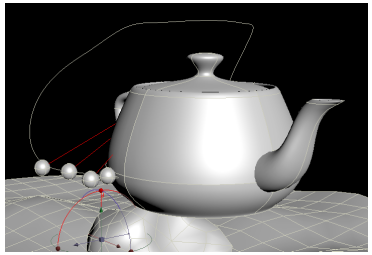
Constrained Points	**fixed_points**



When **PLAY** is pressed, the **fixed_points group** of the **strap base curve** now constrains the dynamic wire to the animation of the teapot.

Under the **Material > Elasticity section** of the **Wire Object DOP parameters**, **RMB** on **all four** of the main **parameters** in turn, and choose **Revert to Defaults** from the resulting menu



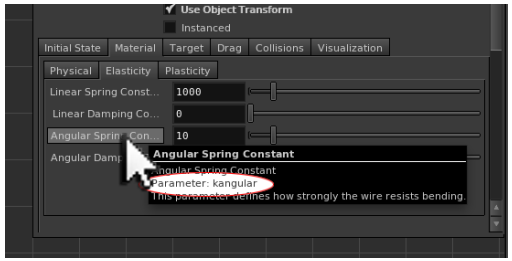


These parameters control the **flex of the dynamic wire**. The **default settings** for these parameter will create a far more springy dynamic curve. When **PLAY** is pressed, the dynamic curve flops around relative to the movement of the teapot and any interaction with the seabed_terrain, animated_teapot and rock dynamic objects.

See file dynamic_strap_stage1.hipnc

CREATING DIFFERENT DYMANICS FOR SPECIFIC AREAS OF THE CURVE

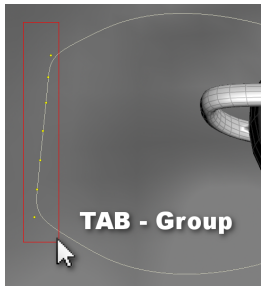
Currently the entire dynamic curve is fully flexible. Portions of the curve can however have bespoke stiffness so that a handle on the dynamic strap can be added.

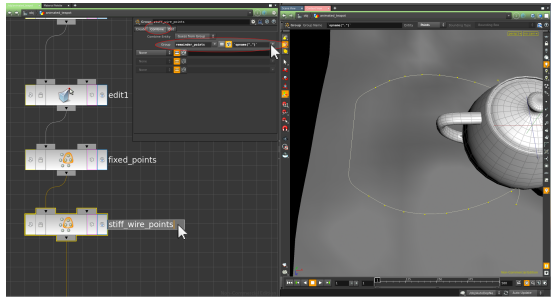


Hover the **mouse over** the **Angular Spring Constant parameter**. The associated **information card** that appears **reveals** its **internal name** as **kangular**. Similarly, hovering the mouse over the **Angular Damping Constant** reveals its **internal name** of **dampangular**. As these parameters controls the wire bend, increasing them will stop the wire from bending.

As with **Material Parameters**, **attributes created on geometry** with the **same internal name** as any **existing parameter**, will **override** the **default parameter value**. In this context, **specific points** of the **dynamic wire** can be **assigned** a much **higher kangular** and **dampangular values** to **prevent them** from **bending** under dynamic forces.

Rewind the simulation, and back **inside** the **animated_teapot object**, select all of the **flat points** of the **base curve** in the **Viewer** and **group** them. **Rename** the resulting **Group SOP** node to **stiff_wire_points**.





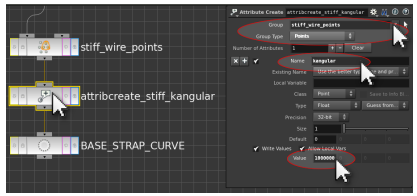
Under the **Combine** section of the **parameters**, specify:

Group **remainder_points** **≠** **`opname(".")`**

This will create a **second group** of **points** that do not equal the stiff_wire_points.

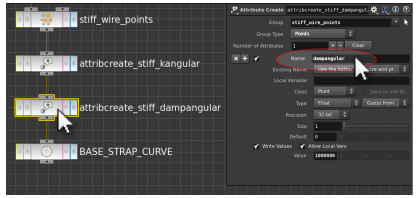
Append to the **stiff_wire_points Group SOP** an **Attribute Create SOP**. In its **parameters** specify:

Group	**stiff_wire_points**
Group Type	**Points**
Name	**kangular**
Value	**10000000**



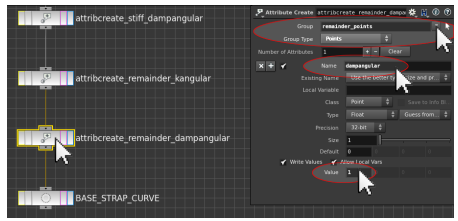
Copy (**CTRL + c**) and **Paste** (**CTRL + v**) this node, and **rename** it to **attribcreate_stiff_dampangular** and wire it beneath the first Attribute Create SOP. In its **parameters** specify:

Name	**dampangular**



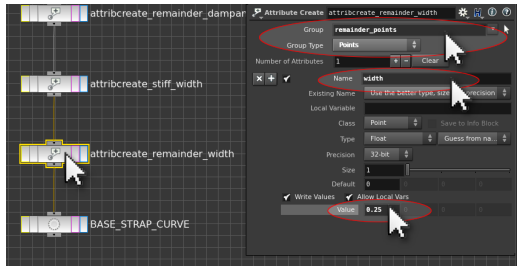
This will assign high kangular and dampangular values to the stiff points of the curve preventing them from bending.

Repeat this process, creating **two more Attribute Create SOPs** that only affect the **remainder_point groups**. Set the **Value** of **attribcreate_remainder_kangular** to **10**, and the **Value** of **attribcreate_remainder_dampangular** to **1**.



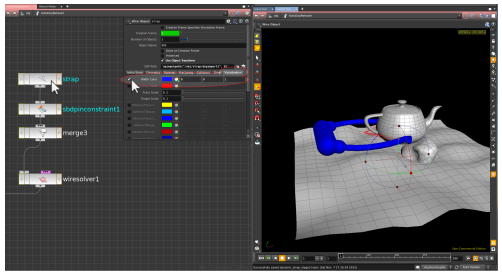
This will set the remainder points of the curve to the default kangular and dampangular values found on the Wire Object DOP.

The **width of the curve** can **also** be **assigned** in a similar way. This will allow the **stiff handle area** of curve to be a **wider collision object** than **the main strap**. A custom **width attribute** will override the **Material > Physical > Width parameter** of the **Wire Object DOP**.



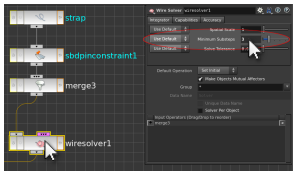
Create **two final Attribute Create SOPs** to control the **width** of the dynamic curve for both the **stiff_wire_points** (**Value 0.85**) and the **remainder_points** (**Value 0.25**).

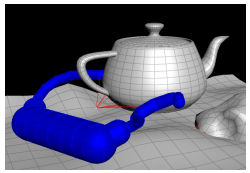
At **DOP Level**, increase the **Material > Physical > Width parameter** of the **Wire Object DOP** to **1**, and under the **Visualization** section of the **parameters activate** the **Width Colour tick box**.



The custom width of the dynamic curve is now visible, and can be used to gauge collisions with the other dynamic objects.

As a **final step**, **increase** the **Minimum Substeps parameter** of the **Wire Solver DOP** to **3**. This will increase the number of collision calculations taking place per frame, and as a result **increase the accuracy** of the simulation.





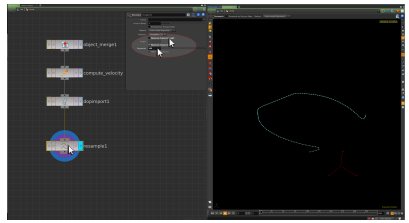
When **PLAY** is pressed the **results of the custom attributes** created can be seen in the dynamic wire simulation.

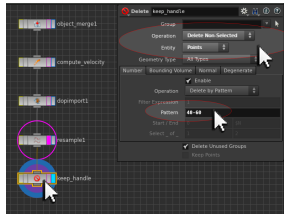
See file dynamic_strap_stage2.hipnc

BUILDING THE STRAP AND HANDLE GEOMETRY

Go **inside** the **strap object**, and **append** to the **dopimport1 node** a **Resample SOP**. This will **rebuild** the **dynamic NURBS curve** as a **Polygon Curve**, also **increasing** its **point count**. In its **parameters** specify:

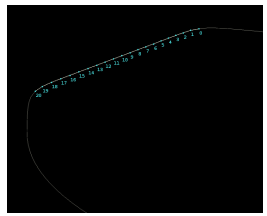
☐	**Maximum Segment Length**
☑	**Maximum Segments**
Segments	**100**





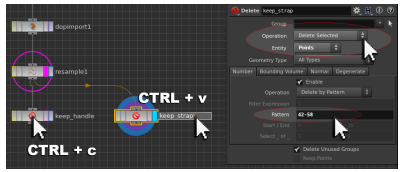
Append a **Delete SOP** to the Resample SOP and **rename** it to **keep_handle**. This will isolate the points creating the handle. In its **parameters** specify:

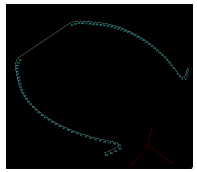
Operation	**Delete Non-Selected**
Entity	**Points**
Number >	
Pattern	**40-60**



Copy (**CTRL + c**) and **Paste** (**CTRL + v**) this node to create a second version of it. **Rename** this **Delete SOP** to **keep_strap** and **modify** its **parameters** to:

Operation	**Delete Selected**
Entity	**Points**
Number >	
Pattern	**42-58**



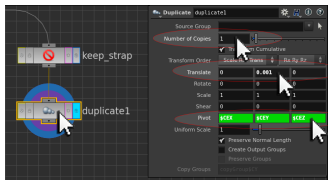


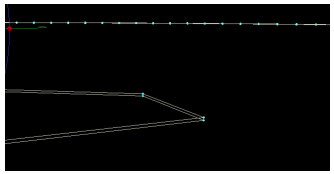
This will remove the handle points from the strap geometry, ensuring that any deforming points of the handle do not interfere with the final effect.

Append to the **keep_strap Delete SOP** a **Duplicate SOP**. In its **parameters** specify:

Number of Copies	**1**		
Translate	**0**	**0.001**	**0**
Pivot	**$CEX**	**$CEY**	**$CEZ**

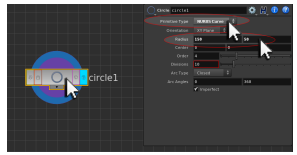
This will create a second instance of the curve, discretely sitting just on top of the original.





This pair of curves can now be used to control a **Rails** operation, where geometry is swept between the rail curves in order to create the strap geometry effect. As a **new network chain**, create a **Circle SOP**. In its **parameters** specify:

Primitive Type	**NURBS Curve**	
Radius	**150**	**50**

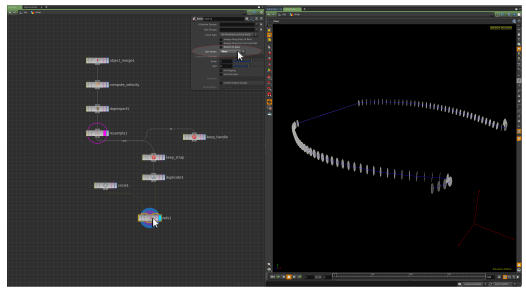


While this is seemingly a very large piece of geometry, its size will get re-ranged relative to the duplicated curves when a Rail operation is performed.

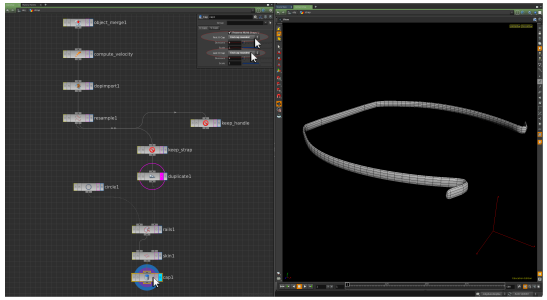
Append to the Circle SOP a **Rails SOP**, and wire the **output** of the **Duplicate SOP** as the **second input**. In the **parameters** for the **Rails SOP** specify:

Use Vertex	**None**

This will create a sweep of the circle geometry along each point of the curve rails.



NOTE: If a Sweep SOP were used to create this effect instead of a Rails SOP, there would be a large amount of twisting on the resulting sweep geometry breaking the illusion of the strap. While the Rails SOP is very similar to the Sweep SOP in terms of its function, it will produce far more stable strap geometry as a result of its internal sweep being controlled by two rails.



Append to the Rails SOP a **Skin SOP**, followed by a **Cap SOP**. In the **U Cap section** of the **parameters** for the **Cap SOP activate rounded end caps**.

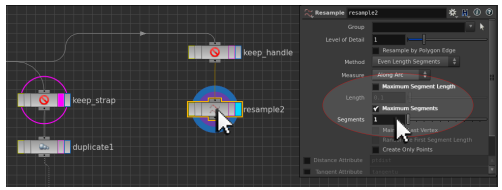
See file dynamic_strap_stage3.hipnc

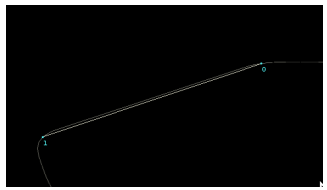
CREATING THE HANDLE GEOMETRY

Append to the **keep_handle Delete SOP**, a **Resample SOP**. In its **parameters** specify:

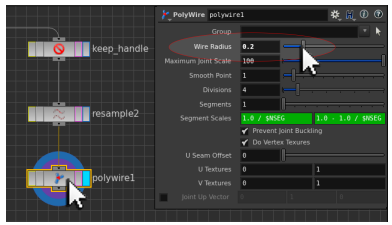
☐	**Maximum Segment Length**
☑	**Maximum Segments**
Segments	**1**

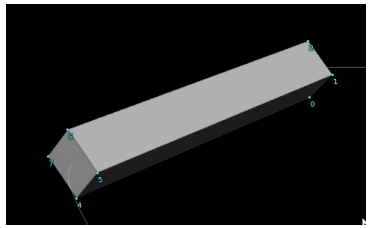
This will redraw the curve, keeping only the first and last point.





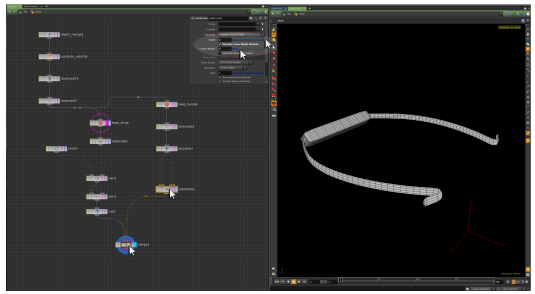
This curve is now a rigid bar that can build the final handle geometry.



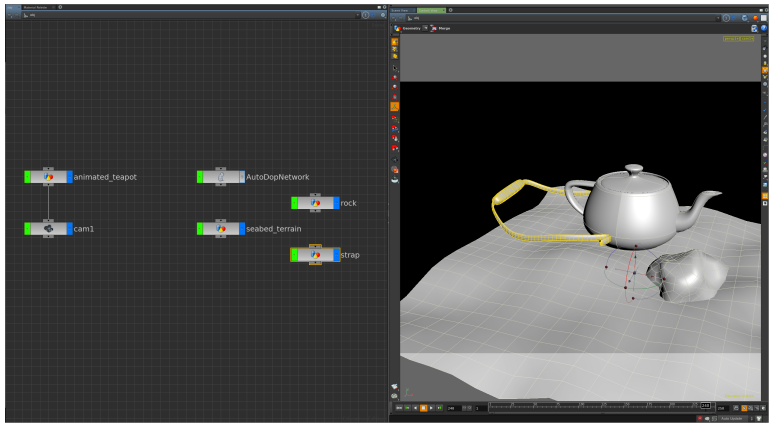


Append to the **Resample SOP**, a **PolyWire SOP**. In its **parameters**, specify a **Wire Radius** of **0.2**. This will create a solid block of geometry for the strap handle.

This handle geometry can also be subdivided to create a smoother handle shape, and merged with the strap network.



Back at **Object Level**, the effect of the **strap** and **handle geometry** can be seen in the **context** of the **dynamic wire simulation**.



The **springiness** of the **dynamic curve** can be further adjusted by **modifying the kangular** and **dampangular Attribute Create SOPs** of the **animated_teapot** to create a more naturalized effect.

The **width Attribute Create SOPs** can be adjusted to **reduce intersections** of the handle and strap geometry with other scene objects.

Similarly, a **Wind Force DOP** can be appended into the AutoDopNetwork after the Gravity DOP to help **increase** the sense of **underwater currents**.

See file dynamic_strap_complete.hipnc