

**RIGMAROLE
STUDIO**<https://rigmarolestudio.com>

Search 🔍 ☰

#character-rigging (/tag/character-rigging/) #face-rigging (/tag/face-rigging/) #rigtip (/tag/rigtip/) #maya (/tag/maya/)

Rigging Spherical Eyes with Just a skinCluster and Sin and Cos

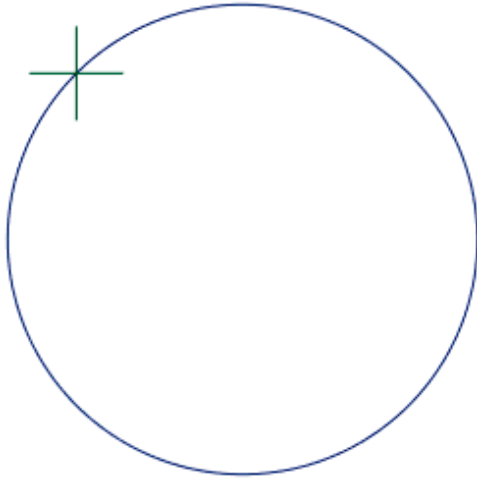
CHRIS LESAGE (/AUTHOR/CHRISLESAGE/) JUNE 14, 2019

This is a quick Maya character rigging tip about how to rig perfectly spherical iris and pupil scaling in your characters, using only a single skinCluster and some basic nodes. I used Maya, but this technique can easily be mapped to all the other 3D applications using their own sine and cosine math functions.

(I'll include a more full tutorial with example scripts and files in the future.)

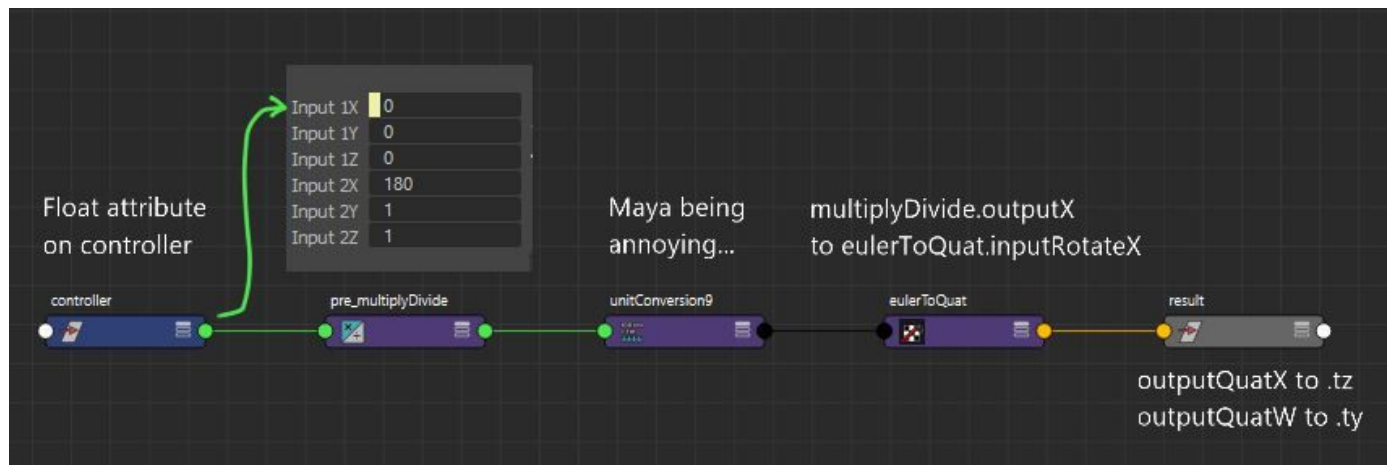
Remember when Karolis Galkus (<https://vimeo.com/galkus>) gave that tip about using eulerToQuat nodes to drive sin and cos (<http://tech-artists.org/t/maya-sine-function/9903/5>) on the TAO forum? Basically, with just a few nodes, using the **eulerToQuat** node, multiplied by 180, you can output a sin or cos wave, using just standard Maya nodes. outputQuatX is sin and outputQuatZ is cos.

If **translateZ** is sine and **translateY** is cosine, now you can drive a point along a circle.

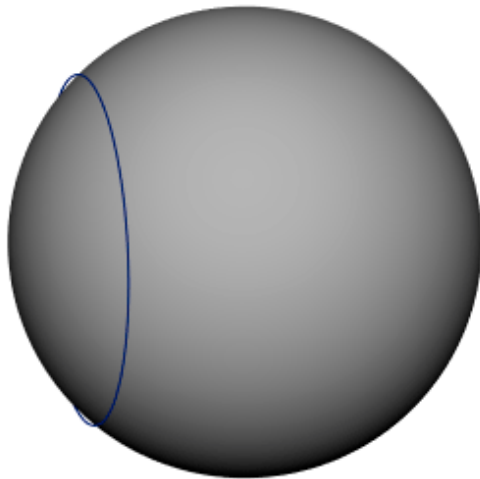


Ok, big deal. Not too interesting yet. You could do that just by rotating too.

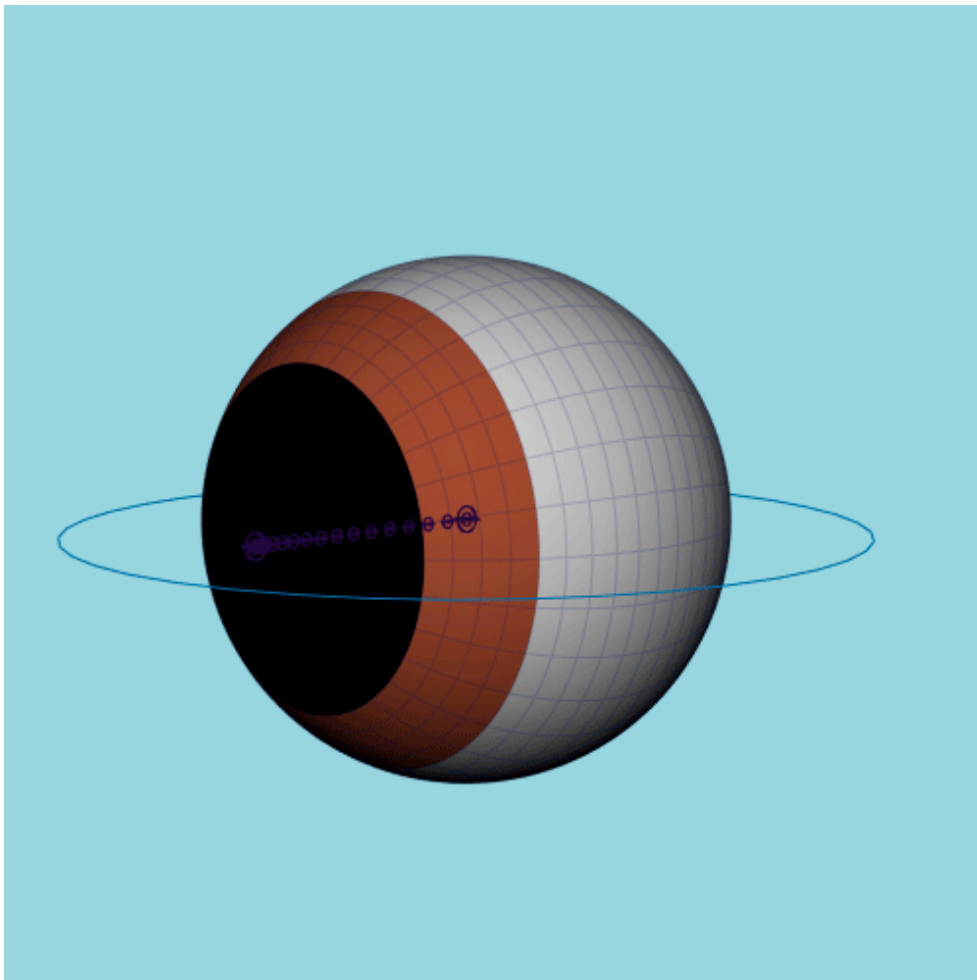
Here's how the basic graph looks for driving this effect:

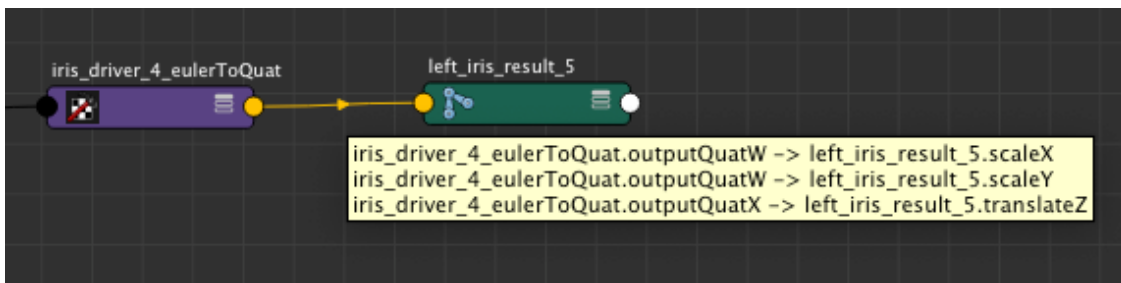


Next, if **translateZ** is sine and **scaleX** and **scaleY** is cosine, you can now drive a circle along a sphere. That's a bit more interesting!



And now, if you drive joints instead of circles, you can skin a polygon sphere, and drive the edge-loops along the surface of the sphere. One joint per edge-loop is all it takes.





And by doing this, you can have a perfectly spherical pupil and iris scaling effect, without the need for perfectly parametric blendshapes or any fancy deformers.

Spherical eyes are usually preferred for very cartoony rigs. And even on realistic rigs, you could still use this sphere as a clean basis. Then you could create the cone and bulge effects of the eye by using shaders, or an additional deformation or joint offset, on top of the perfect sphere.

There are ways to create this kind of iris and pupil effect with lattices and other deformers. But this technique is 100% skinCluster, and so it is also perfectly game-engine friendly.

A few extra footnotes about this:

- If you export to a game-engine, you may need to scale your joints in X, Y **and** Z. Some game engines do not support non-uniform scaling. As long as your joints line up with your edge-loop, scaling in Z should not cause any distortion.
- Instead of eulerToQuat, you could use maya-math-nodes plugin by Serguei Kalentchouk (<https://github.com/serguei-k/maya-math-nodes>). Personally, I freelance with many different clients. And in this last year I've had to support Maya 2015 to 2018. (Still avoiding Maya 2019, for various bugs...) So when I can solve something with built-in nodes, without the need for extra plugin dependencies, that's a win for me, for my clients, and for my rigs to survive long-term through Maya upgrades.

- Yes you could write your own Maya plugin to do sin and cos. But Maya's plugin API is a vicious beast that I'd rather not sacrifice my life to. ͇(͇)͇ Also, I like to write tutorials that beginners or intermediate rigging artists can also follow.
- You can also use sin and cos functions in Maya expressions. 10 to 15 years ago, expressions simply meant slow rigs. And so I think many of us built up a fear of using them. But on modern computers, sometimes they are fine to use. They do seem to punch a big giant hole in parallel processing... So profile and test! I still prefer nodes in Maya.
- Other 3D apps like Cinema4D, Houdini, and Blender either have math nodes, or integrated ways to add basic math formulas to your rigs. So this technique would work in any of them. Cinema4D has Xpresso, Blender has a sine node, and Houdini likely has 1000 different ways to do this. A simple VEX snippet or VOPs, channel expressions, constraint networks, CHOPs if you're brave... etc. Alas, here we are stuck with Maya. :)

Thanks for reading!

Share this article

 LinkedIn

 Pinterest

 Facebook

 Twitter

(http://www.linkedin (http://pinterest.cc
(https://www.facebook (https://twitter.com mini=true&url=http url=https://rigmaro
u=https://rigmarol text=Rigging%20Sf spherical- spherical-
spherical-eyes/) spherical-eyes/) eyes/&title=Rigging eyes/&description:
(/author/chrislesage/)



Chris Lesage is a freelance tech artist, character rigger and procedural artist. He's worked on productions like The Little Prince, Majora's Mask Terrible Fate, Tom Clancy's The Division and more.

🐦 @chrislesage (<https://twitter.com/chrislesage>)

📍 Montreal, Canada

Sign up for blog updates and tech art tips

Subscribe

🐦 (<https://twitter.com/chrislesage>) 📷 (<https://www.instagram.com/rigmarolestudio/>)

in (<https://www.linkedin.com/in/chrislesage>) 📡 (<https://rigmarolestudio.com/rss/>)

© 2019 Rigmarole Studio

