

Facial Rigging Tutorial for 3ds Max v.8

By Tonya Payne


<http://www.rigging3d.com>

This facial rigging setup utilizes bones and a user interface inspired by Jason Osipa's *Stop Staring* facial rig. For those of you use to Maya, this is a little different because Max lacks things that Maya has, such as a Channel box and Expression editor. Nevertheless you can accomplish the same type of rig. This tutorial is not an intro to rigging, but instead more for people who already know how to rig but don't know how to go about setting up a bone based facial rig in Max. If you learned to rig in Maya, this is also for those of you who want to/have to use Max and want some translation between the two programs. Since Maya and Max use different terminology, words found between <> are Maya words. All other terminology is Max specific, unless I happen to slip up! If you have never set up a similar rig before, I would recommend skimming the ENTIRE tutorial first so you'll see where you're going. It will save you time and frustration.

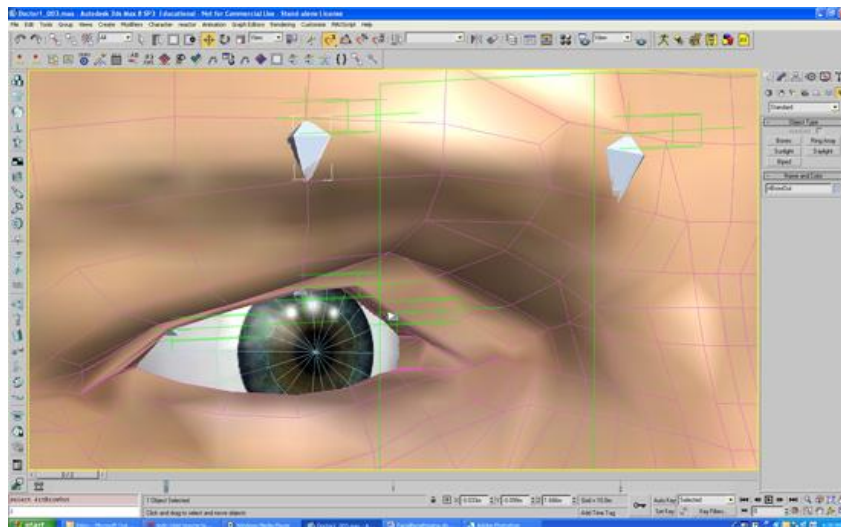
****I've left some notes since I wrote this when I didn't understand Max very well, they'll be highlighted blue and will correct some misunderstandings that I had about how things work****

Placing Bones <Joints>

Before placing your bones:

1. You will need to set your mesh <geometry> to where you can see the wireframe on shaded by pressing F4.
2. You will also need to turn on the 'snap to vertex' option. Right click on the icon on the top tool bar that looks like a magnet with a three next to it. This will bring up Grid and Snap Settings. Select Vertex and uncheck anything else. Close the window. Make sure that the icon is yellow by left clicking on it. 
3. Now you are ready to place your bones. From the Main Menu, Create→Systems→Bones IK Chain
This can also be found on the roll out menu on the right hand side of your screen by clicking the first icon (an arrow) and then clicking the last icon below that (gears). Notice that on the right a Parameters menu opens and you can alter the size of the bones if you need to. You will probably have to make them small to fit around the eyes and in the corner of the mouth. It will depend on the size of your model.
4. The bones should snap to the vertex. Select the vertex you want them on then click on it. Then click on a vertex next to that one to complete the bone, then right click to end the process. You can delete the end bone that Max automatically creates, you don't need it. This is how you will place the bones around the eyes, the brow, the mouth and any bones you may place for the nose and cheeks.

5. For the Jaw bones, you will use a bone chain, starting at the base of the ear and placing roughly three bones for each side of the face to the side of the chin.
6. After you have finished placing all of the bones, open the Schematic View <Hypergraph> so you can see what you just placed. If you double click on the nodes slowly you can rename them here. Name all of your facial bones using whatever naming convention you are used to.
7. Next hide the mesh by right clicking on it and selecting Hide Selection. This is so you can see the bones you've placed better.
8. Now you're going to make helpers to control your bones.
9. From the Main Menu, select Create→Helpers→Point
10. On the rollout menu that comes up on the right side of your screen, make sure that Cross and Box are selected. You may need to alter the size as well to match the size of your model.
11. Place your Helpers to match up with the beginning point of your bones (the first vertex you clicked on). The snap should still be enabled so they'll snap to the bone.
12. In the Schematic View, rename all of your helpers to match up with the bones they're associated with, adding something like Helper or Controller to differentiate in the name.
13. To Unhide your mesh, on the rollout menu on the right click the second to last icon (looks like a TV) and scroll down to the Hide menu. Select Unhide by Name. That will open a box. Select your mesh by name and click Unhide.



An example of how the bones and helpers look

A User Interface

Your User Interface can be almost anything you want it to be, keeping in mind functionality and ease of use. Mine is based entirely on Jason Osipa's User Interface. You will want to plan out in advance what you are wanting out of your Interface. Basically the Interface consists of Squares or Rectangles that have Circles in them. The Circles in the end will control the facial bones indirectly. Each direction will have a Custom Attribute associated with it.

1. First you will want to switch to the Front Orthographic view by pressing the F key.
2. Then from the Main Menu, Create→Shapes→Rectangle
3. Hold the CTRL key to keep the shape Square. Drag the Mouse and make one shape. Drag it again and make another Square within the first one.
4. Then from the Main Menu, Create→Shapes→Circle (these can also be accessed from the rollout menu on the right side of your screen)

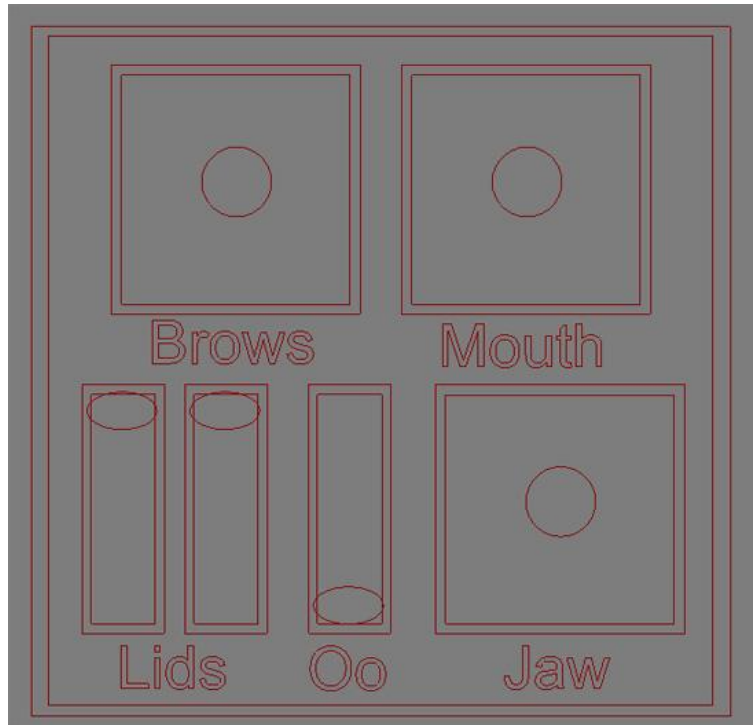


5. Draw a circle to sit inside the Squares
6. Name the Circle something associated with the set of facial joints it will be controlling, for example, Brows or Mouth.
7. Parent the Circle to the Inner Square and the Inner Square to the Outer Square. This is important for later, because the Circle derives its local coordinates from the parent.
8. You can now duplicate these into as many as you need.
9. There is another type of controller you can make for your interface. This one is for Custom Attributes that only need to move in one direction, such as Blinking.



10. Do the same as above but scale your shapes to look like this:

11. Make all of the shapes you need and then make text to name them.
Create→Shapes→Text In the Rollout menu on the right, you can change the size and what the text says.
12. After you've named all of your shapes on your interface you can create two large rectangles to go around them. Parent all of the small outer rectangles to the big inner rectangle and it to the outer big rectangle so they are all in a hierarchy. Group them, name the group UIGroup. Also, to change the color, select the whole Interface and click the color box in the rollout menu.



A sample (and simple) UI. Depending on how much movement you need, the UI could be very complex and have asymmetrical movement. For example the Brows would be separate. This one is symmetrical with control over both at once. The good thing is with a bone based setup, the animator can also animate directly on the face and is not tied to just the user interface unlike with Morph Targets <Blendshapes>.

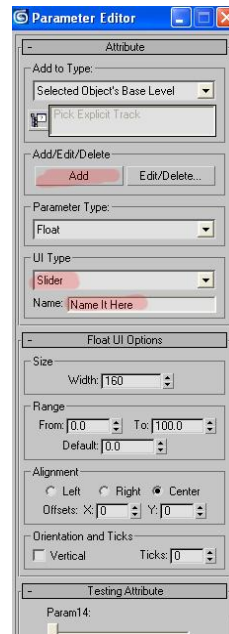
Custom Attributes

Custom Attributes in Maya function a little differently than the ones in Max, in that they are added to the Channel Box. In Max, they are added to the Parameter Rollout. Make sure you have them in the order you want them before starting to connect them using the Reaction Manager. Once you have them setup, the Reaction Manager uses their location, not their name, as what it's hooking things up to. If you move them around afterwards, you'll have problems. Save often!

We are going to add our Custom Attributes to one of the Rectangles around the User Interface. You could add them to anything, but I decided this would be best, since the Rectangles aren't going anywhere.

1. Select one of the Rectangles and name it UserInterface.
2. Select the UserInterface. Animation → Parameter Editor
3. Change UI type to Slider and the Name to the name of the Custom Attribute you're adding. Plan them out in advance. In this case I'm doing:

BrowUp
BrowSqueeze
MouthOpen
MouthClose
Smile
Frown
UpperLip
Oo (this is where the lips
pucker out)
JawOpen
JawLeft
JawRight
BlinkLeft
BlinkRight




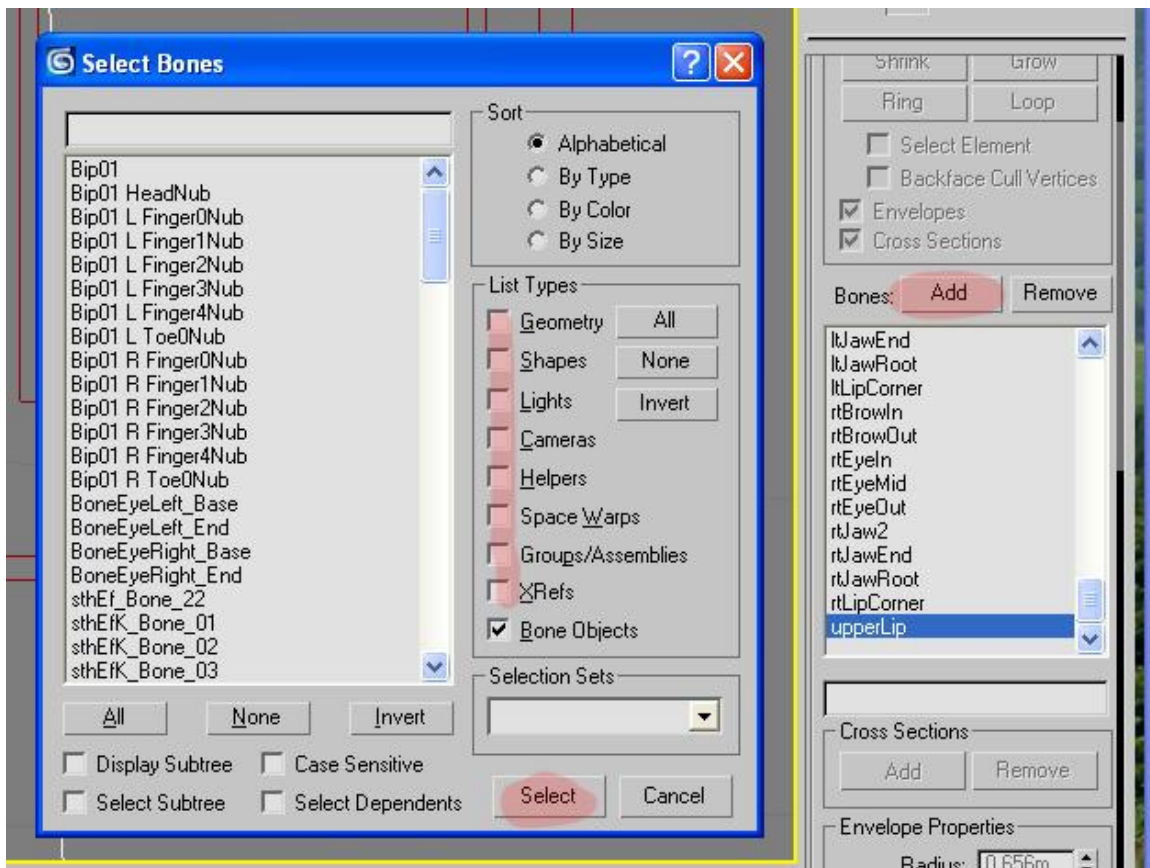
4. To add them, click the Add button. You have to do them one at a time.

Skinning

Now, before we can start making face shapes with our bones, we have to skin them to the mesh. Max skinning is a little different than skinning in Maya in the way it is done.

1. Parent each of the bones to their respective helpers. For example the leftJawRoot would be parented to the leftJawHelper and the leftBrowIn would be parented to the leftBrowInHelper. The helpers should now control the bones.
2. Then group all of the helpers into one group, for example call it facialGroup, then this group needs to be parented to the head bone.
3. Select the mesh.

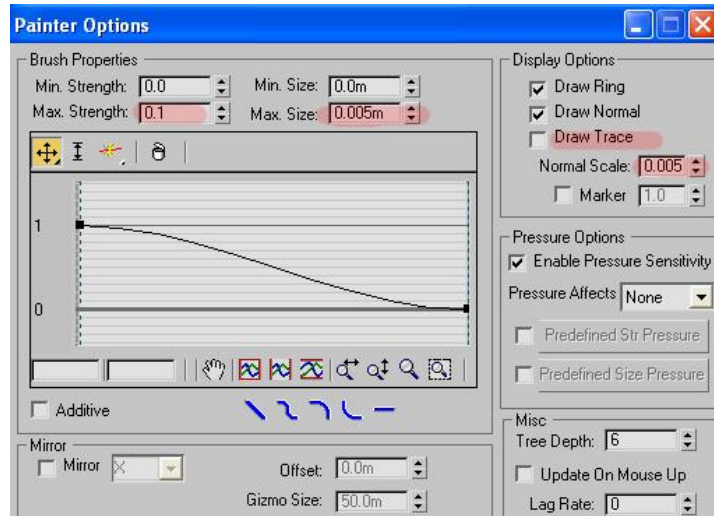
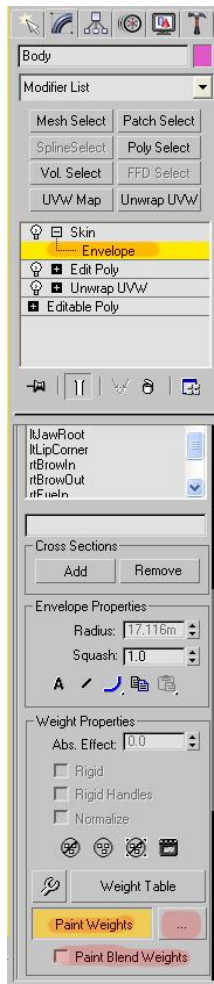
- 4 In the rollout table on the right, select the Modify button, it's the second from the left and looks like part of a gray tube 
- 5 Under Parameters there is a list of bones that are skinned to the mesh, if you have already skinned your character. Otherwise the box is empty. Either way you're going to add the face bones to the mesh (or all the bones if you haven't skinned yet). Click 'Add.'
- 6 A window pops up called Select Bones. Select the bones you want to add. You can uncheck boxes from the list type to help you narrow it down to bones only. When you've highlighted all of the bones you want, click 'Select'



Here is the box that pops up when you click the add button above the list of bones that are skinned.

- 7 Now grab some of your helpers and see how the weights are affecting the mesh. You will have to paint some weights more than likely to get the face to deform the way you want it to. Painting weights is better in Max in that there is color information that is accurate as to the weight on each vertex, with blue being close to zero and red being 1.

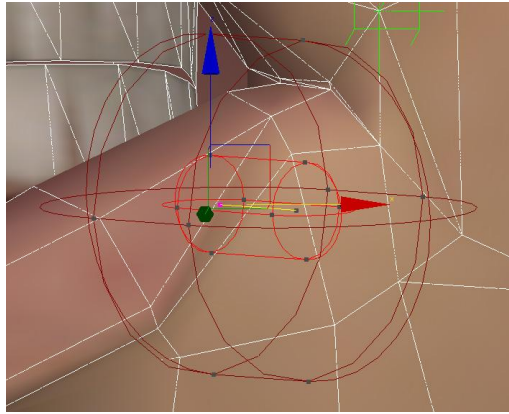
- 8 To paint weights or alter them manually through the weights table, make sure your mesh is still selected and that you are still on the 'Modify' tab. Select 'Envelope' from the list of things affecting the rig. This enables you to alter the weights.
- 9 Uncheck 'Paint Blend Weights'
- 10 Click Paint Weights. A window should pop open. If it doesn't click the button with the ellipses. This will bring up the paint weights window.



The settings here will depend on your scene size. The scene size that I'm using is to scale to the real world and is thus very small in Max.

- *Max Strength* should be low so that you add weights a little at a time. If it's negative, weights are removed.
- *Max size* is dependent on your scale. The size you use should be relative to what you're working on. For example, the size will be very small when working around the eyes, but bigger if you're painting weights on the shoulders.
- I turn off *draw trace* as a matter of preference. Draw trace leaves a trail in red as you paint that disappears when you let up on the mouse. I don't like it but you may.
- *Normal scale* is the size of the normal that points out from the brush. I make it small because I don't like seeing it either.
- You can also adjust the fall off rate by adjusting the curve in the table.

Tip: You may have to adjust the size of the envelopes. Max uses envelopes that are red wire spheres or tubes that determine default weight. To alter them, select them and use the move <translate> gizmo to make them smaller. The best case scenario if you're adding bones is that there will be no weights on the bones you added and you'll have to add weights to them.



An envelope, generally it will be near the bone, but sometimes Max screws up and it's huge.

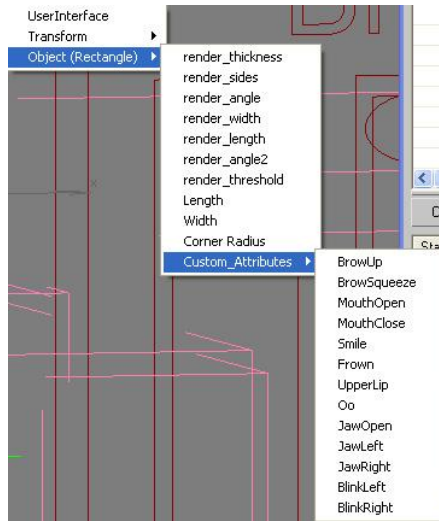
Tip: It may help to set keyframes on your helpers on frame one and then turn on auto key and move to frame two. Now you can move your helpers around all you want and you only have to return to key one to return them to their default location. Once you're done you can delete the keys.

Reaction Manager <Set Driven Keys Wannabe>

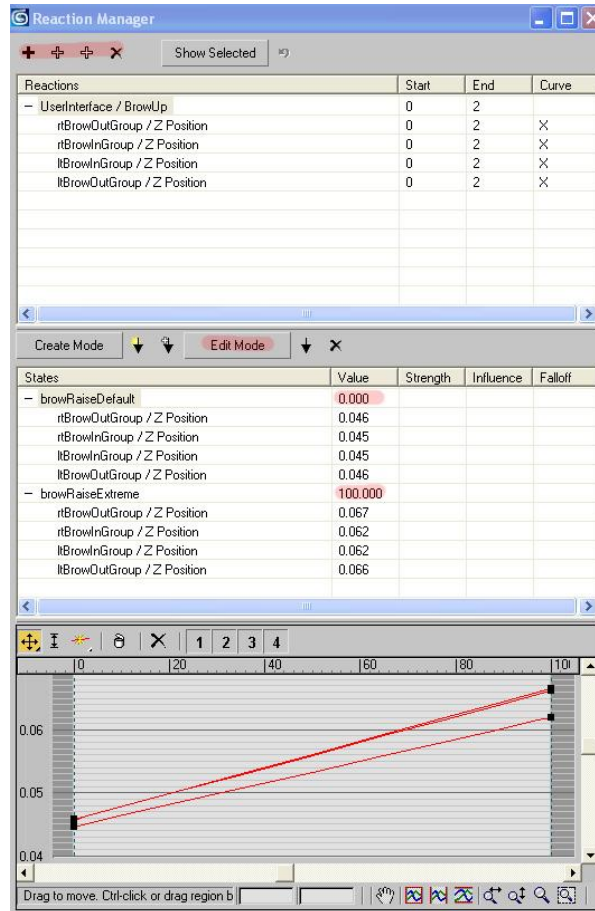
Once you have the weights roughed out where you want them, you can begin linking your helpers to your Custom Attributes. To do this, you're going to need some null groups. You cannot have more than one master<driver> for each slave<driven>. This is sort of like the Set Driven keys in Maya, but not as easy to use. Group each helper. You will need a group for each helper that you need to connect to a Custom Attribute.

Tip: If you need to add additional groups after you've connected things in the Reaction Manager, group the helpers themselves, not the groups, because local space for the hierarchy is apparently determined by the node highest in the hierarchy. If you parent the groups that are connected in the Reaction Manager, your helpers will go wacky.

- 1 Open the Reaction Manager, Animation→Reaction Manager
- 2 Click the solid black plus sign (Add Master) and then click on the rectangle that you have stored all of your Custom Attributes on. This brings up a menu, follow the menu to your Custom Attribute and click on it.



- 3 Once a Master is in the Reaction Manager, select the Slaves that will be controlled by this Custom Attribute. The easiest way is to select them in the Schematic View <hypergraph> because you're selecting groups. Once you have selected them you can add them by clicking the plus sign that is white with a black border. A similar menu to the one above will come up and you will load the channel that will be controlled, for example the Y Position.
- 4 Once you have both of these loaded, a state in the lower window is automatically created. You can rename this state to something like customAttributeDefault. Then right click anywhere in that lower window and select "Create State"
- 5 This adds another state, which you can name something like customAttributeExtreme.
- 6 Click the button that says "Edit Mode"
- 7 Change the number on the second state (extreme) to 100. This represents the Custom Attribute Slider being at 100. Then you can select your groups that are being controlled, if they're not already selected. Use the move tool to move the groups in the channel that is the slave. Notice that the values change in the boxes.
- 8 Now select the rectangle that has your Custom Attributes on it. Under the Modify tab is where your attributes are located. Slide the slider to 100 and you should see the helpers move the mesh.



Tip: If you right click, you can “Append State” and this will update all the channels into the state that is in the lower table.

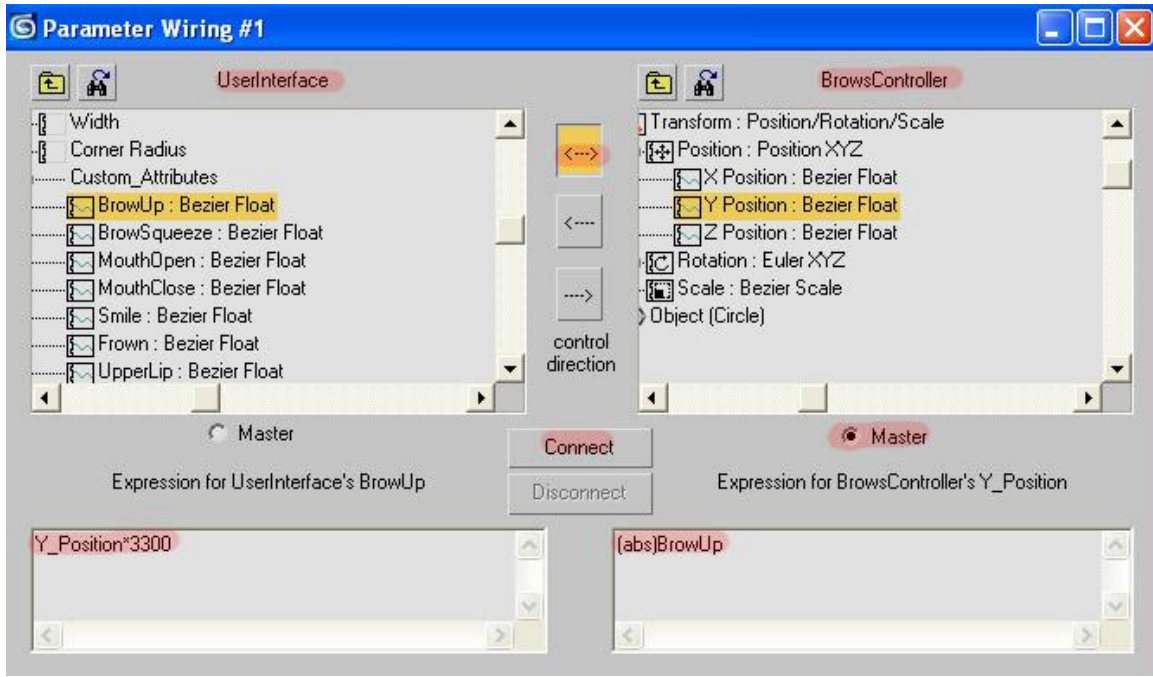
Wiring the User Interface <Expression Editor Wannabe>

So, in Maya, to wire <connect> your attributes to your User Interface, you would use the Expression Editor and write some very simple mathematical scripts. It’s sort of a similar method in Max, except it’s called Wire Parameters. Wire Parameters will work fine for this, although they do not seem to have the same breadth of functionality as the Expression Editor.

- 1 First you are going to want to limit which axes the circle controllers can move on. Click on the Hierarchy tab in the rollout menu. It’s a little box with three other boxes connected under it. Then select ‘Link Info’ and check the boxes on which axes and channels that you don’t want the object to be able to move.

Tip: Sometimes for some reason, an object appears to be moving on one axis, for example the Z axis, and so you lock the other two, but when you try to move it again, you find out that it’s really moving on the Y. I’ve had this happen with the controllers for the UI. ****This is because it’s local space that counts, not world space.****

- 2 Also, before you start, you're going to need to group the circle controllers in the UI so they're translation settings will be reset <freeze transformations, but without being able to zero them out in a channel box, if you want to be able to return them to their default positions, you will have to set keys on them> ****Actually, you can switch to parent space, and this will allow you to zero them out****. You then need to use them, not the group, to connect to the Custom Attributes.
- 3 Right click on the rectangle that is storing the Custom Attributes and select Wire Parameters. After selecting your channel as you did when you used the Reaction Manager, a wire appears and you want to next select the first circle controller that you want to connect to a Custom Attribute. (You may want to hide the group markers so that you don't accidentally select them. To do this, right click on the group markers, the little brackets around what is grouped, and select 'Hide Selected.')
- 4 After selecting a channel there, a box will pop up showing what you've chosen to connect.
- 4 On the left will be the User Interface and on the right will be the controller (make sure it's the controller and not the group, you won't have to worry about this if you've hidden the group markers). You want to select the two way arrow and make the controller the Master by clicking on the circle.
- 5 Next you want to alter the script in the boxes to make your controller control the Custom Attribute Slider from 0 to 100 while moving from the top to the bottom (or left to right) of the rectangle that marks it's border. You can multiply or divide the Position by a number to do this. It will depend on the size of your scene. My scene is extremely small so I multiply by a large number. With the script box on the right, you will want to use (abs) to make the number always positive. If you want the negative side of the Position to be controlled then you will make the script on the left side negative, for example:
 $-Y_Position * 3300$
- 6 Click "Connect" or if you've already connected and just altered something, "Connect" will be replaced by "Update"



- Now if you move the controller in the direction that you had wired it to the Custom Attribute with, you will see that the facial bones associated with it move.

Tip: If your facial bones go wacky, make sure that you have grouped all of the circle controllers, otherwise, this will not work because ungrouped they are using world space coordinates. If you screw up by say, wiring the wrong thing and have to disconnect it you may have to unlink the circle controller from it's hierarchy. Unlink <unparent> the controller in the Schematic view by right clicking on it and selecting 'unlink', delete the group that you had made for it, and then link it back to the rectangle and then group it again. This clears out all stray numbers.

Tip: You can connect up to four Custom Attributes to a controller that is square/bidirectional and one or two Custom Attributes to a controller that is rectangular/mono-directional. You have the choices of positive and negative Y and positive and negative X.

Once you've finished wiring all of your circle controllers to the Custom Attributes they will control, you're nearly finished.

You will next want to limit the movement on your controllers so they don't go outside the rectangles that mark its border. You must do this after you have wired them to the Custom Attributes or you will not be able to wire them properly. It will make it easier on the animators if you put a limit controller on them.

- To do this, select the circle controller and go to the Motion tab in the rollout menu, it looks like a moving wheel. You want to click on the first tab that says 'Assign Controller' and expand it. Then find the channel that you want to limit, in this case I select the Y Position.

- 9 Click the little box above the list with the little question mark in it and add a controller. This opens up a window. You want to select "Float Limit" and click OK.
- 10 Another window opens. Both Upper and Lower Limit should be Enabled. Adjust the numbers until the circle can't go outside of the rectangles that it's in. Once more this depends on the size of your scene. I used .035 and -.04 for my scene. The numbers may be different if the circle wasn't dead center when you started, and that's okay.
- 11 Now, you want to freeze <template> the interface so the animators won't have to work around the rectangle borders. Even if you have locked their movement through the "Link" option, you can still select them. Select each one and right click then select "Properties" and uncheck 'Show Frozen in Gray.' This allows them to keep their color. Click OK.
- 12 Next right click again and select 'Freeze Selection' Now you can't select them unless you right click and select 'Unfreeze All'

Tip: If the object disappears when you freeze it, CTRL-z to undo and go to your Display (TV icon) tab in the rollout menu, and make sure that the 'Hide Frozen Objects' under the Hide menu is unchecked.

Once you limit all of your controllers, you're pretty much done! Now all that is left is to touch up weights on the face so that it deforms how you want it to.

~~~~~

I created this walk through tutorial after needing a bone based facial rig in Max and not being able to find much online that was useful. I was trained in Maya and when I started this, Max was a foreign concept to me. It took me nearly a week to tear it all apart and come up with this rig. It took me mere hours to redo the rig and create this walkthrough. I hope this tutorial helps you. If you find any mistakes or think I've done something a little weirdly and you think you have a better way, please email me and let me know:  
tonya@rigging3d.com