Summary

6.1 The purpose of a control rig is to simplify the animation process, making it easier for animators to pose their characters in a 3D environment.

6.2 Animation, by definition, is the ability to bring something to life through movement.

6.3 Kinematics is the study of motion.

6.4 FK is a method where a hierarchy of joints or objects is rotated into a position and keyframed at every point of the hierarchy.

6.5 IK is a mathematical system that calculates the rotations of joints in a predefined chain.

6.6 When animating, IK provide the ability for stickiness.

6.7 Maya has three different IK solvers: the SC solver, the RP solver, and the spline solver.

6.8 Since IK solvers have a tendency to break and stop solving, a control system is used so that keyframes are not lost during animation.

6.9 It is common to create multiple control chains for different areas of the body for maximum control.

6.10 Maya uses an Euler rotation method when calculating the rotations of joints during an animation. When evaluating a joints rotation from one position to the next, the rotations are considered by a specific order dependent on the X, Y, or Z axis.

6.11 Gimbal lock occurs when two of the joint axes align during a rotation causing the inability to rotate in a particular direction.

6.12 It is important to animate in Gimbal mode so that you know exactly what type of rotations are occurring.

6.13 The connection editor is a tool that allows you to create a direct connection from one attribute to another.

6.14 The expression editor provides a place where MEL programming language can be used to create mathematical expressions within the scene environment.

6.15 Set Driven Key is a powerful tool unique to Maya that provides attribute control using keyframes set on one or more attributes as they relate to the value of the attribute in control.

6.16 Constraints are a leader and follower relationship, where one object's attribute leads while another object's attributes follows. Constraints can be keyframed on and off.

6.17 A point constraint controls the translation values of an object.

6.18 An orient constraint controls the rotation values of an object.

6.19 A parent constraint controls both the translation and rotation values based on the world axis.

6.20 An aim constraint controls the rotational values of an object and points them toward another object using an aim vector.

6.21 A pole vector constraint controls the pole vector values of an RP IK solver.

6.22 A geometry constraint constrains the translation position of an object to the surface of another.

6.23 A normal constraint constrains the rotation orientation of an object to the surface of another.

6.24 A group node is an empty transform node, or a transform node above one or more objects in a hierarchy. Think about group nodes as invisible boxes that can be empty or have one or more objects inside.

6.25 Group nodes can be placed above any object when constraints are used.

6.26 Clusters are deformers that provide the ability to control CVs or vertices. In character setup, they are used to help control the Spline IK solver.

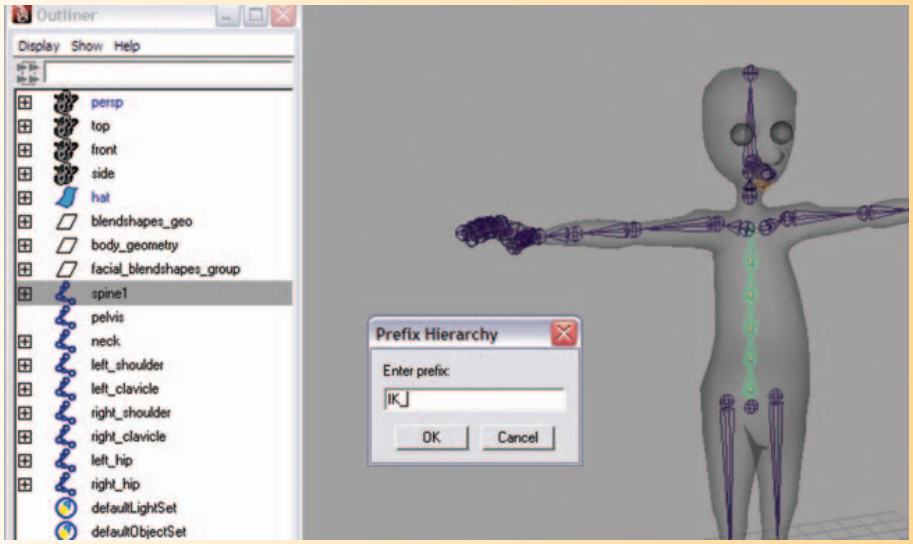
Assignments

Assignment 6.1: Creating a Control System for the Spine Skeleton

Most of the controllers used in these assignments are made from the NURBS circle. In the companion DVD, I have provided some different shapes that can be used instead. If you would like to use the control shapes that I have provided, simply go to **[File > Import]** and find the Maya file on the DVD to import it into your scene file.

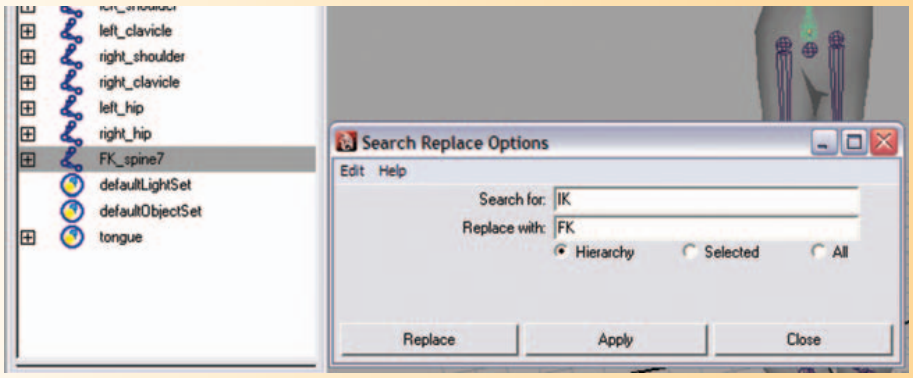
Set up your work environment by doing the following:

1. Open Maya and set your project.
   - From your computer's desktop, go to **[Start > Programs]** and **select** Maya.
   - Once Maya is open go to **[File > Project > Set...]** and browse to your project folder then **click OK**.
2. Open your last saved file: Go to **[File > Open]** and **select** *05_asgn05.ma*.
3. Continue working in X-ray Joints mode.
4. Make sure that your geometry is placed on a layer and that the layer is set to **R** for reference so that you are unable to **select** the geometry by mistake when working.
5. To make selection easier, open your outliner by going to **[Windows > Outliner]**. (In order to control the spine and still keep it flexible and natural looking, we will create an FK chain that controls an IK chain.)
6. Create an FK joint chain and an IK joint chain for the spine by doing the following:
   - **Select** the *spine1* joint.
   - Rename the original chain by adding the IK_prefix. Select **[Modify > Prefix Hierarchy Names...]** and set the following:
     i. **Enter** prefix: "**IK_**".
     ii. **Click OK**.



Adding the IK prefix to the spine joint chain.

- Duplicate the IK chain by going to **[Edit > Duplicate]** or press **(ctrl+d)**. (The duplicated joint chain begins with IK_*spine7*. Since there already is an *IK_spine1* joint that begins the original joint chain, Maya places the next number as to not label with the same joint name. Having two *IK_spine1* joint chains would cause confusion in Maya. Maya will not use the exact name for multiple nodes. However, be careful NOT to use the exact name for multiple nodes when the nodes are in different hierarchies.)
- **Select** the *IK_spine7* joint chain and rename the hierarchy by going to **[Modify > Search and Replace Names...]** and set the following:
  i. **Search** for: "IK".
  ii. **Replace** with: "FK".



Renaming the duplicated IK chain as an FK chain using search and replace.

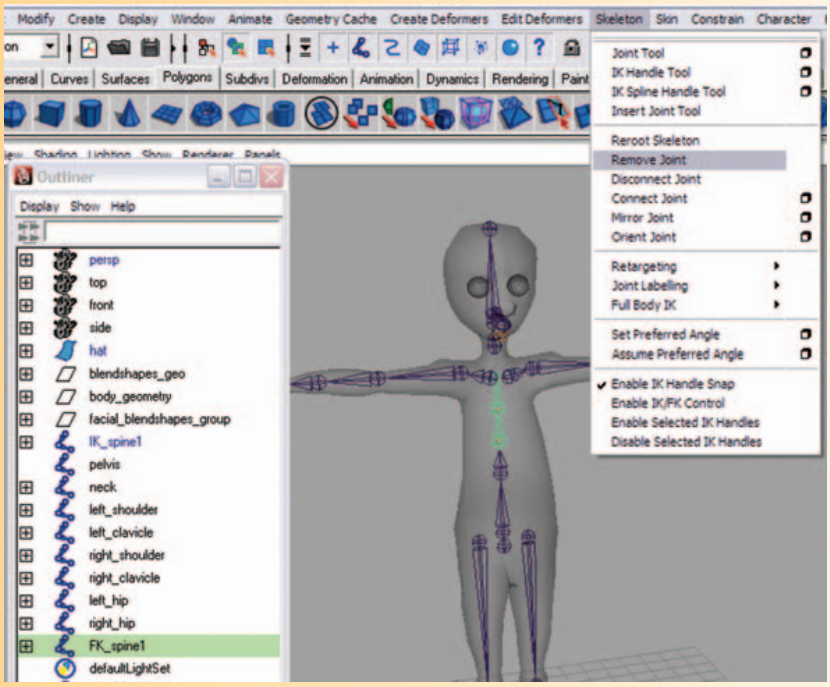
- **Select** *FK_spine7* and in the channel box, rename it to *FK_spine1*.

You should now have two joint chains for the spine – one FK chain (beginning with *FK_spine1*) and one IK chain (beginning with *IK_spine1*).

7. We will now create the FK spine first. Remember, the FK spine should be created first so that the correct rotational orders can be set. It is very helpful to HIDE the *IK_spine1* joint chain so that it does not get in the way while creating the FK controls.

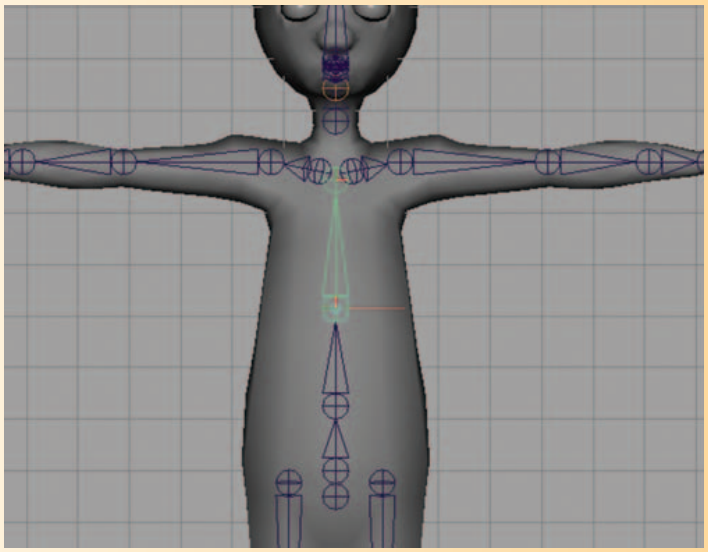
> **!** To hide something in Maya, **select** the object and press **(ctrl+h)**. You will notice that it turns blue in the OUTLINER. To make it visible again, **select** the object in the OUTLINER and press **(shift+h)**.

8. In the OUTLINER, **select** the *IK_spine1* joint chain and press **(ctrl+h)** to hide the chain.
9. For the FK chain, we will not need all six joints – four should be plenty. We can remove the extra joints from the FK chain one at a time:
   - **Select** the *FK_spine2*.
   - Go to **[Skeleton > Remove Joint]**.
   - **Select** the *FK_spine5*.
   - Press the **(g)** key to repeat the last command of removing a joint.



Removing extra joints in the FK spine.

10. **Select** the move tool by pressing **(w)**, press the **(insert)** key on the keyboard, and **reposition** the *FK_spine3* joint on so that it is level at the waist of your character. Reposition the *FK_spine4* joint (up or down) so that it is level at the base of where your character's rib cage would be located.

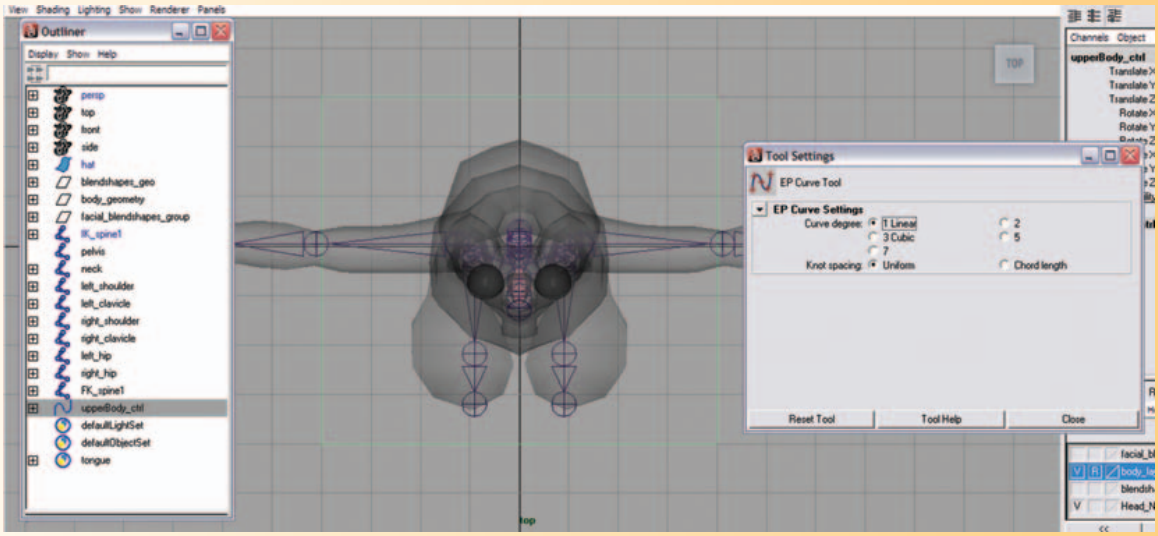


Repositioning the FK spine joints.

11. Create a NURBS control for the upper body by doing the following:

- Go to **[Create > EP Curve Tool – option box]**.
- Under *EP Curve Settings*, **change** the following:

  Curve degree: choose "1 linear".
- In the top view, use the grid snap tool by holding the **(x)** key and **click** to **draw** a square around the body of your character. Hit **"enter"** when completed.
- In the channel box, **rename** the square *upperBody_ctrl*.



Creating the upperBody_ctrl.

- In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *FK_spine1* joint, and **click** the **MMB** (middle

mouse button) and drag it slightly to snap the *upperBody_ctrl* into place. (This is an easy way to ensure that the pivot of the square is snapped precisely in place. It might not seem easy at first, but after you have done this a few times, it will be easy. The important thing to remember is to place your mouse cursor over the joint that you want the controller to control. Do not **click** on the move tool when using the snap key.)

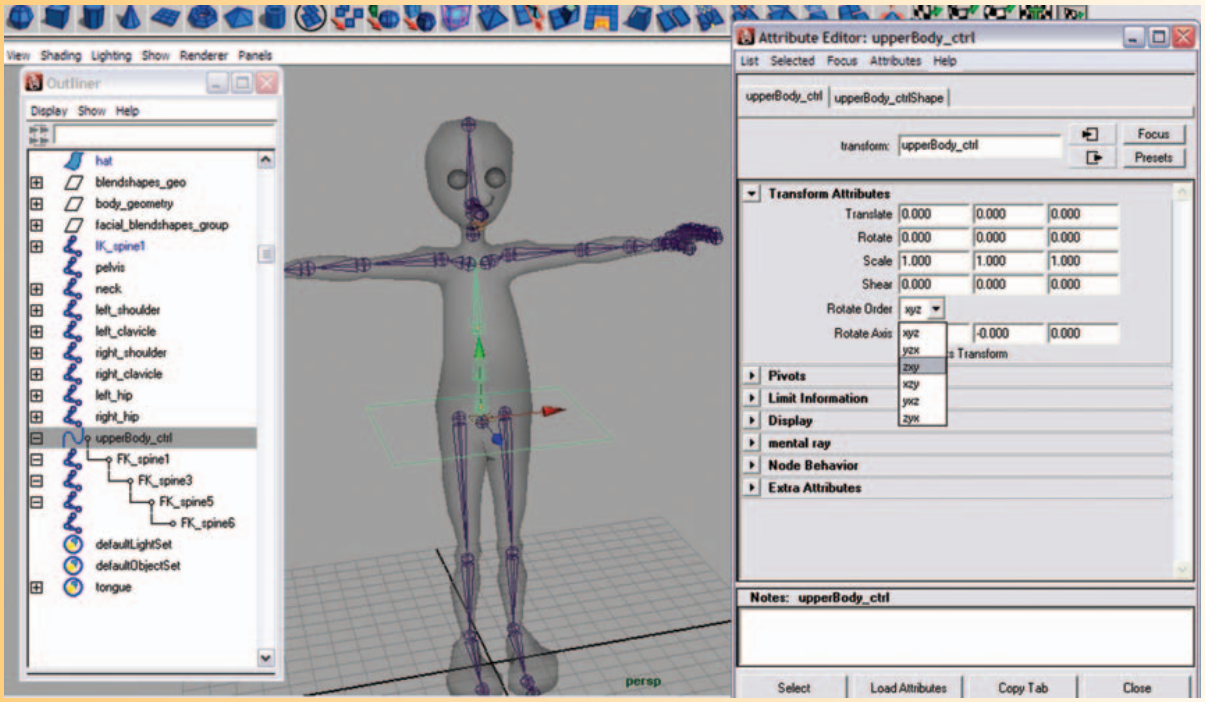
- Use the scale tool by pressing **(r)** and **resize** the square if necessary. (This control should be scaled large enough that it is far OUTSIDE of the character's geometry to make it easy to **select**.)
- With the *upperBody_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return the translate values to 0 and the scale values to 1.)
- In the OUTLINER, hold down the **MMB**, **click** on the *FK_Spine1* joint and **drag** it onto the *upperBody_ctrl*. (This makes the *FK_Spine1* child to the *upperBody_ctrl*. *By moving or rotating this controller, the entire Upper Body will move with it.*)
- **Change** the rotation order for the *upperBody_ctrl* by doing the following:

  i. With the *upperBody_ctrl* selected, **open** the attribute editor by pressing **(ctrl+a)**.

  ii. **Select** the *upperBody_ctrl* tab.

  iii. Under *Transform Attributes* **set** the following:

  1. Rotate order: choose "**ZXY**".



Parenting the FK_spine1 to the upperBody_ctrl, then changing the rotate order to ZXY.

12. **Create** controls for the FK spine by doing the following:

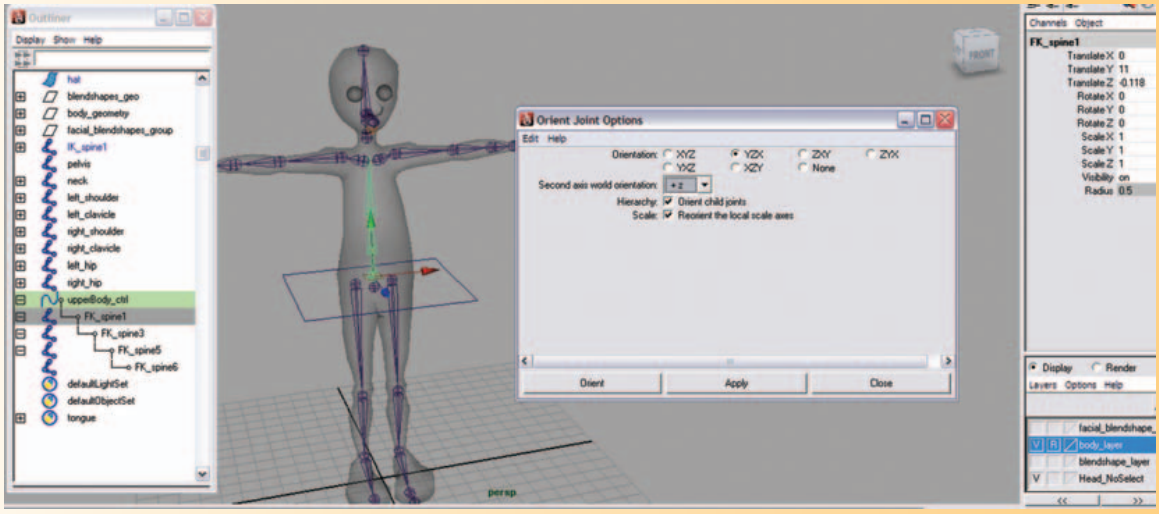
- First, we must reorient the FK spine by **selecting** the *FK_spine1* joint and going to **[Skeleton > Orient Joint – option box]**. **Set** the following:

  i. Orientation: choose **"YZX"**.

  ii. Second axis world orientation: choose **+Z**.

  (This will ensure that the joint orientation aligns with the world axis, which is necessary for FK control.)

  iii. **Click Orient**.

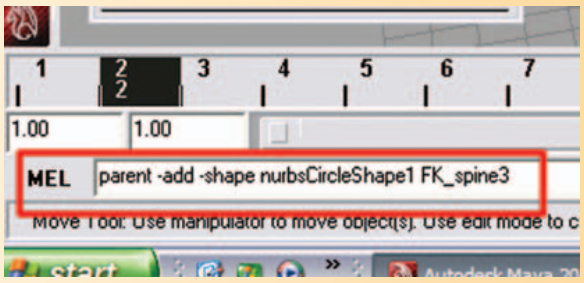


Reorienting the *FK_spine1* joint chain.

- Go to **[Create > NURBS Primitives > Circle]**.
- In the MEL command line, **type** the MEL script below:

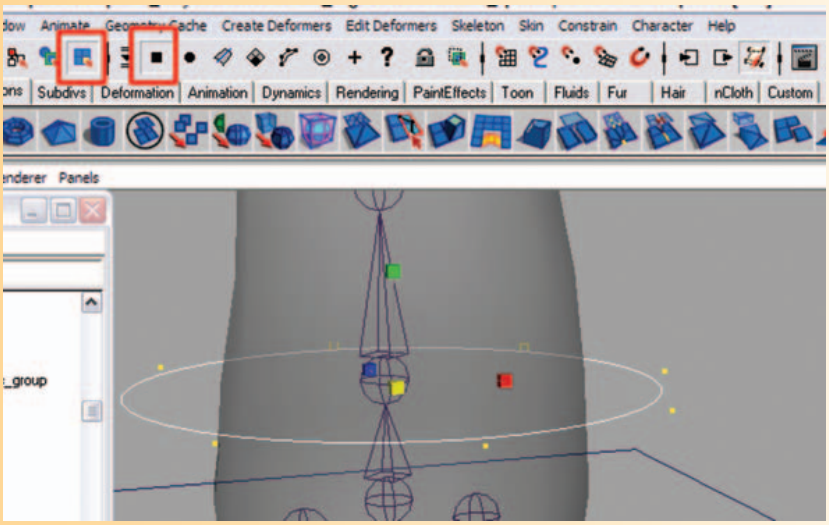
  parent -add -shape nurbsCircleShape1 *FK_spine3*;

  (This command gives a NURBS shape to the joint, making it easier to **select** during animation. It will appear that there are two NURBS circles in the scene.)



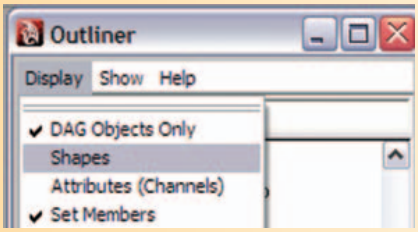
Typing a MEL script in the MEL command line.

- **Resize** the circle to fit around your character's geometry by doing the following:
  i. Press the **(F8)** key.
  ii. Choose the "**select** point components" button in the Status Line.
  iii. Using the scale tool by pressing **(r)**, **click** and drag around the points of the circle and scale them larger so that it extends beyond the character's body to make it easier to **select** when animating. (This affects both circles visible.)



Scaling the nurbsCircleShape1 to fit around the character's geometry.

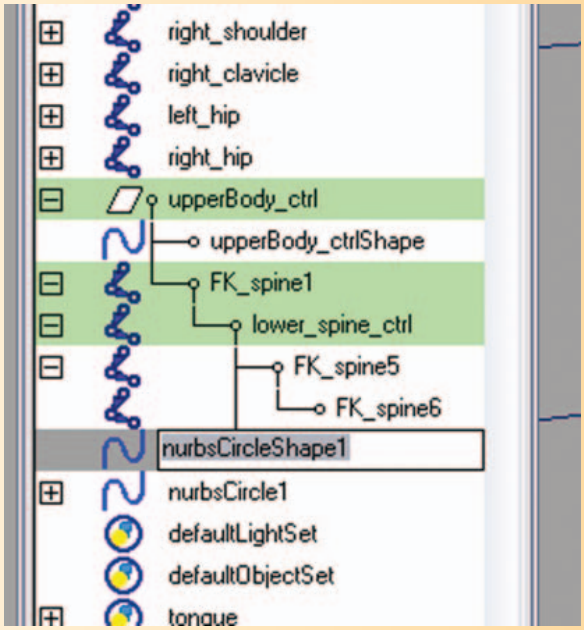
- Press the **(F8)** key to go back into object mode.
- **Select** the *FK_spine3* joint, and in the channel box, **rename** *FK_spine3* to *lower_spine_ctrl*.
- **Change** the rotation order for the *lower_spine_ctrl* by doing the following:
  i. With the *lower_spine_ctrl* selected, open the attribute editor by pressing **(ctrl+a)**.
  ii. **Select** the *lower_spine_ctrl* tab.
  iii. Under *Transform Attributes* **set** the following:
    1. Rotate order: choose "**ZXY**".
- In the OUTLINER, go to **[Display]** and make sure there is a check mark next to *shapes*. If not, **click** on the word *shapes*.



Displaying the shapes in the OUTLINER.

- In the OUTLINER, hold down the **shift key** and **click** on the plus sign **(+)** next to the *upperBody_ctrl* to open the hierarchy and display the children.
- **Double-click** on *nurbsCircleShape1* and **rename** it *LowerSpineShape*. (We must rename the *nurbsCircleShape1* so that Maya does not get confused if we create more NURBS circles later.)

! Feel free to turn the shape option off again later, as displaying shapes can clutter up the OUTLINER and make it confusing when trying to find objects. Just remember to turn it back on when necessary to **select** shapes.



Renaming the *nurbsCircleShape1* to *LowerSpineShape*.

- In the OUTLINER, **select** nurbsCircle1 hit the **delete** key. (We no longer need the NURBS curves – we only needed the shape node.)

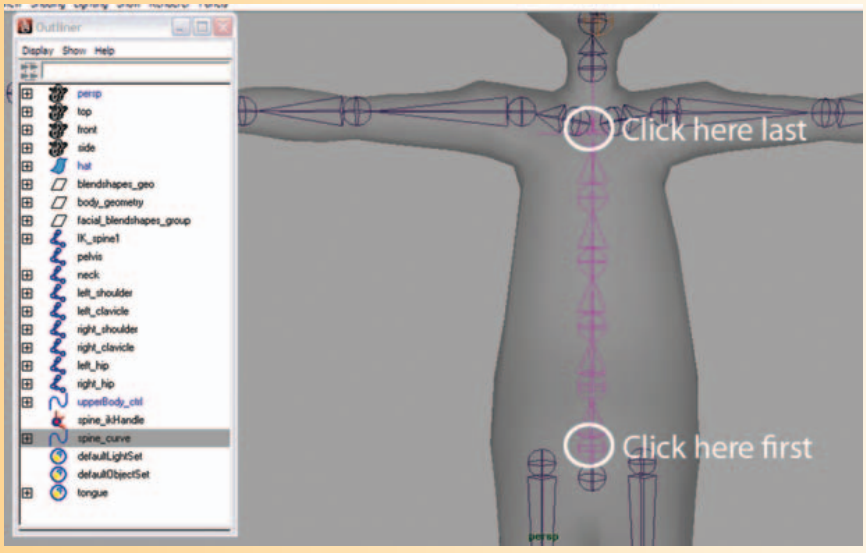
13. **Repeat** this process for the next joint in the spine by doing the following:
    - Go to **[Create > NURBS Primitives > Circle]**.
    - In the MEL command line, **type** the MEL script below:

      parent -add -shape nurbsCircleShape1 FK_spine4;

    - **Resize** the circle to fit around your character's geometry by doing the following:
      i. Press the **(F8)** key to go into component mode.
      ii. **Choose** the "**select** point components" button in the Status Line.
      iii. Using the scale tool by pressing **(r)**, **click** and drag around the points of the new circle and **scale** them larger so that it extends beyond the character's body to make it easier to **select** when animating.
    - Press the **(F8)** key to go back into object mode.

- **Select** the *FK_spine4* joint, and in the channel box, **rename** *FK_spine4* to *upper_spine_ctrl*.
- **Change** the rotation order for the *upper_spine_ctrl* by doing the following:
  i. With the *upper_spine_ctrl* selected, open the attribute editor by pressing **(ctrl+a)**.
  ii. **Select** the *upper_spine_ctrl* tab.
  iii. Under *Transform Attributes* **set** the following:
  1. Rotate order: choose "**ZXY**".
- **Double-click** on *nurbsCircleShape1* and **rename** it *UpperSpineShape*. (We must rename the *nurbsCircleShape1* so that Maya does not get confused if we create more NURBS circles later.)
- In the OUTLINER, **select** nurbsCircle1 hit the **delete** key.

14. Now that we are finished with the FK spine, we can hide it to work on the IK spine. In the OUTLINER, **select** the *upperBody_ctrl* and **press (ctrl+h)** to hide it.
15. In the OUTLINER, **select** the *IK_spine1* joint chain and press **(shift+h)** to display the chain.
16. **Create** the IK spine by doing the following:
    - Go to **[Skeleton > IK Spline Handle Tool – option box]** and **click "reset tool"** then **click** close.
    - In the PERSPECTIVE window, **click** on the *IK_spine1* joint (the bottom spine IK joint) to define the start of the IK joint chain then **click** on the *IK_spine6* joint (the top spine IK joint) to define the end of the chain. (An IK handle appears at the end of the chain.)



Creating the *spine ikHandle*.

- A Spline IK system is created with a curve running through the selected joints. You can then control this joint system by **select**ing the CVs of this curve and moving them. It is awkward, however, to **select** individual CVs (as CVs can only

be selected in component mode) and keyframe them (as CVs only have positions not based on the X, Y, and Z coordinates). We will create cluster deformers on each CV to make them easier to **select**, move, and keyframe on the X, Y, and Z coordinate systems.

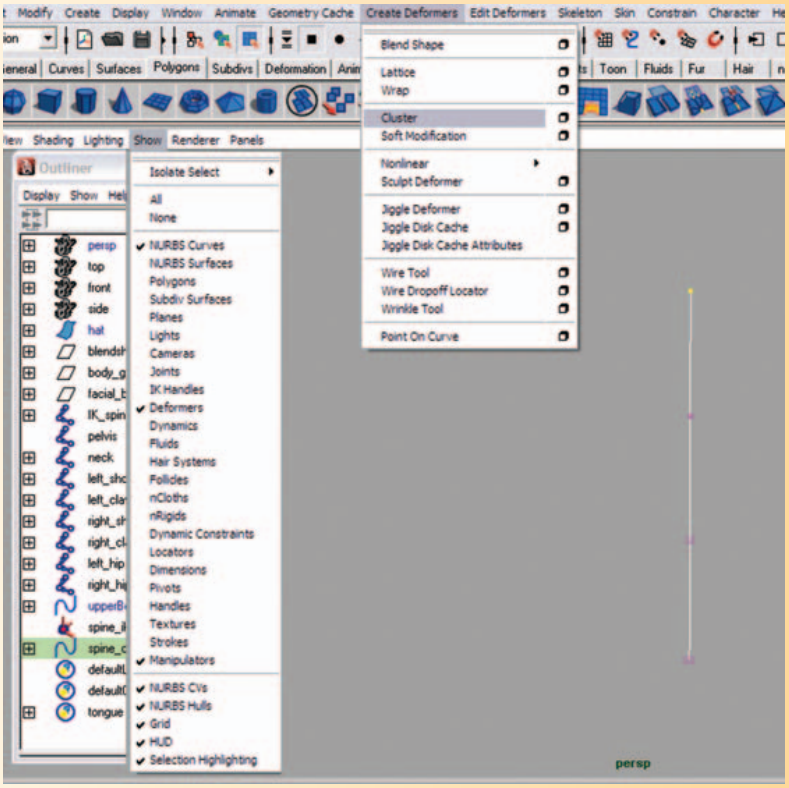
! As you place an IK handle, you can adjust the display size to see it better by doing the following:

Select [Display > Animation > IK Handle Sizes...].

Adjust the slider so that the IK handle is an appropriate size.

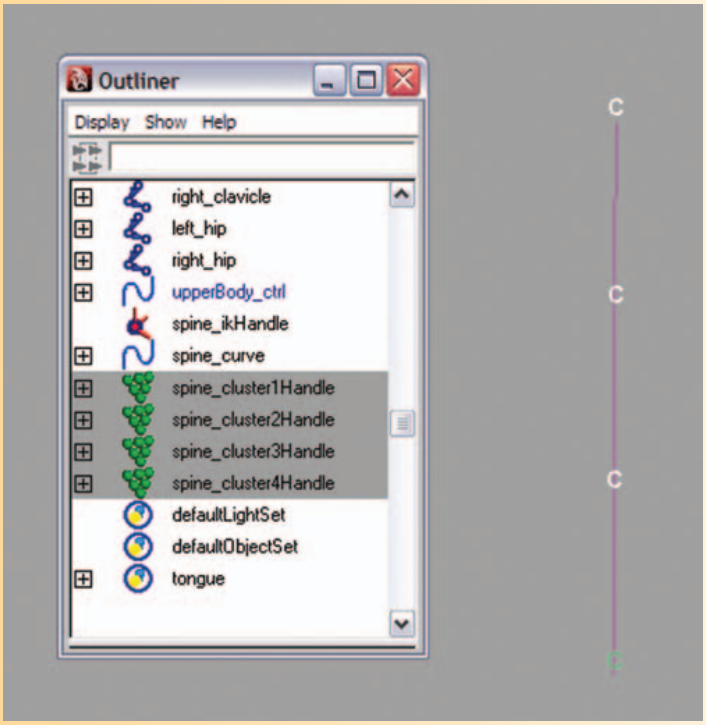
- In the OUTLINER, **double-click** on *ikHandle1* and **rename** it *spine_ikHandle*.
- In the OUTLINER, **double-click** on *curve1* and **rename** it *spine_curve*. (Curve1 is the spline curve that controls the IK solver.)

17. **Create** a control system for the IK spine by doing the following:
    - In the PERSPECTIVE view panel, go to **[Show > None]**, then **[Show > Curves]** and **[Show > Deformers]**.
    - In the OUTLINER, **select** the *spine_curve*.
    - Change to component mode by pressing **(F8)**. Make sure that the "**select** point components" button is depressed in the Status Line.
    - **Select** the top CV point by **click** dragging across the spine to first display the CVs, then **click** dragging around the top CV point.



Selecting a CV on the *spine_curve*.

- **Select [Create Deformers > Cluster]**. (A green highlighted "C" will appear at the top of your FK spine joint chain. If it is not at the top, it usually indicates that more than one CV was selected. If this is your situation, simply undo by pressing **(z)** and reselecting only the top point.)
- **Select** the next point on the curve. Remember, if you are having trouble seeing it, first **select** the curve by **click** dragging across the spine to first display the CVs, then **click** dragging around the second CV point from the top.
- With the second CV selected, press the **(g)** key to repeat the last command of **[Create Deformers > Cluster]**.
- **Repeat** this process for the third and fourth CV points.
- In the OUTLINER, **double-click** on *cluster1handle* and **rename** it *spine_cluster1handle*.
- In the OUTLINER, **double-click** on *cluster2handle* and **rename** it *spine_cluster2handle*.
- In the OUTLINER, **double-click** on *cluster3handle* and **rename** it *spine_cluster3handle*.
- In the OUTLINER, **double-click** on *cluster4handle* and **rename** it *spine_cluster4handle*.



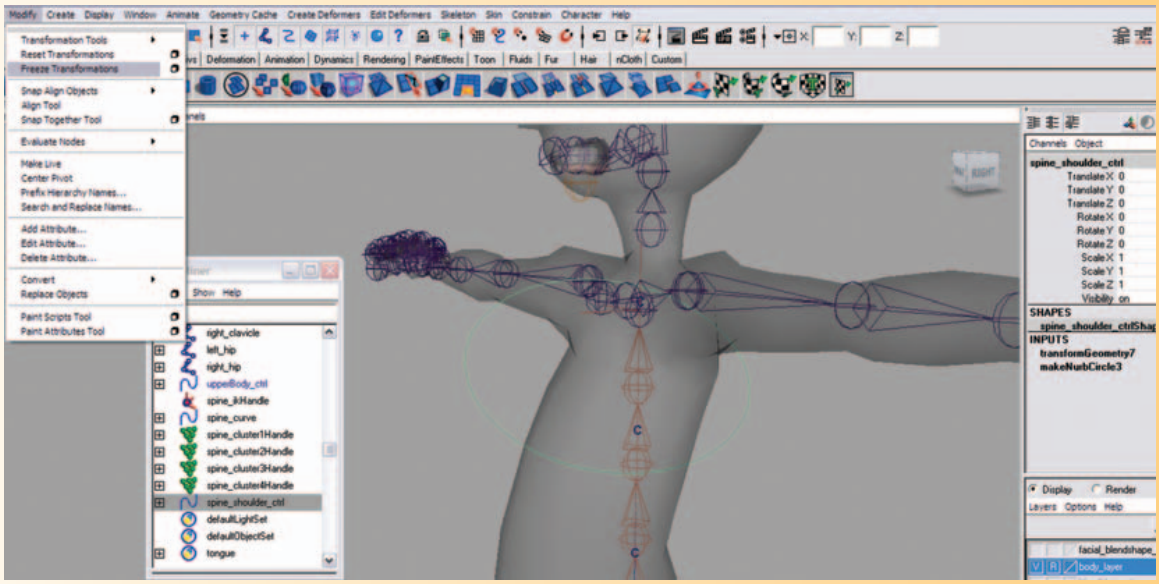
Creating the clusters on the *spine_curve*.

- In the PERSPECTIVE view panel, go to **[Show > All]**.

! It is still difficult to **select** clusters. To make control of the clusters easier, we will parent them to NURBS controllers.

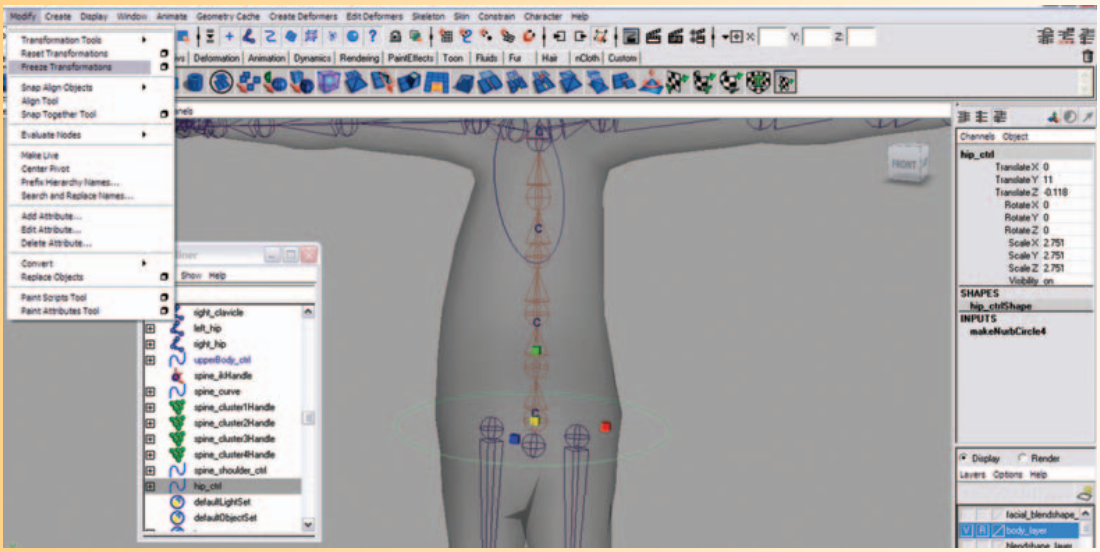
- Go to **[Create > NURBS Primitives > Circle]**.
- In the channel box, **rename** the circle *spine_shoulder_ctrl*.

  Set the **RotateZ** channel to "90".
- In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *IK_spine6* joint, and **click** the **MMB** and drag it slightly to snap the *spine_shoulder_ctrl* into place.
- Use the scale tool by pressing **(r)** and **resize** the circle if necessary. (This control should be scaled large enough that it is far OUTSIDE of the character's geometry to make it easy to **select**.)
- With the *spine_shoulder_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)



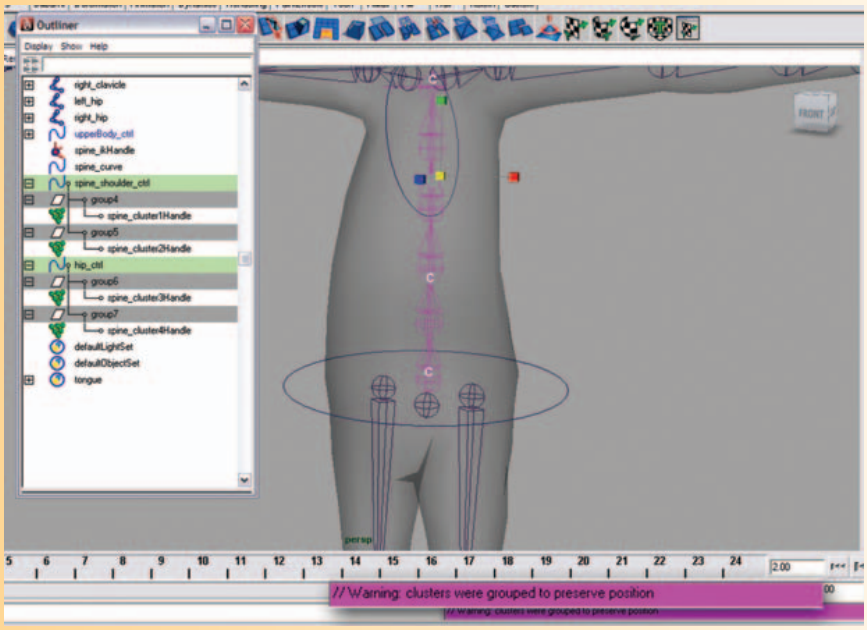
Creating the *spine_shoulder_ctrl*.

- Go to **[Create > NURBS Primitives > Circle]**.
- In the channel box, **rename** the circle *hip_ctrl*.
- In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *pelvis* joint, and **click** the **MMB** and drag it slightly to snap the *hip_ctrl* into place.
- Use the scale tool by pressing **(r)** and resize the circle if necessary. (This control should be scaled large enough that it is OUTSIDE of the character's geometry to make it easy to **select**.)
- With the *hip_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)



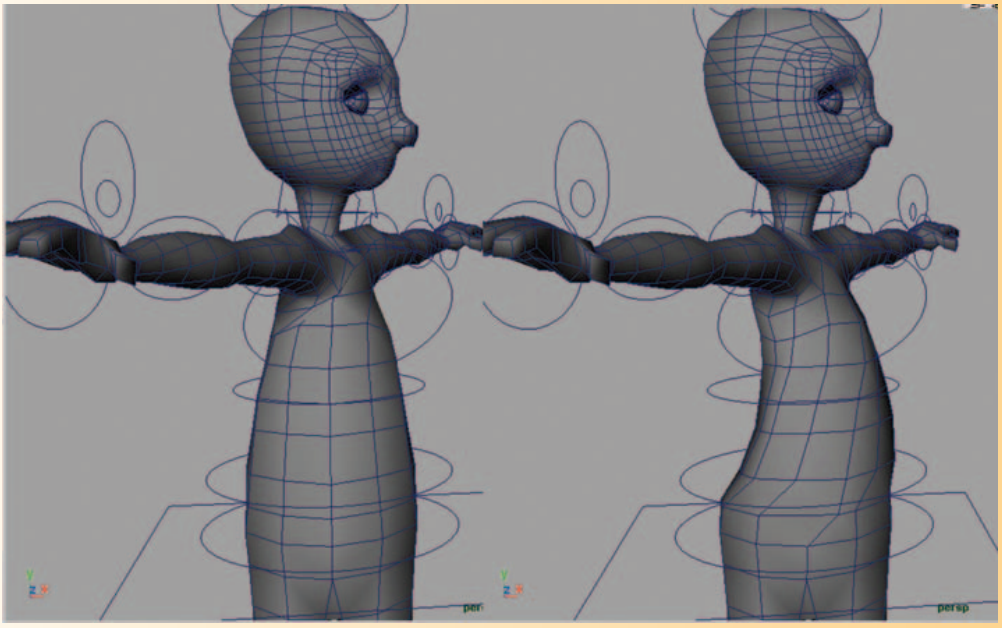
Creating the *hip_ctrl*.

- In the OUTLINER, **click** on the *spine_cluster1handle* cluster; then holding down the **ctrl key**, **click** on the *spine_cluster2handle* cluster and the *spine_shoulder_ctrl*. Press the **(p)** key to parent them. (This makes the *spine_cluster1handle* and *spine_cluster2handle* child to the *spine_shoulder_ctrl*. When you parent the clusters, they are automatically grouped. Maya displays a warning for notification, and is nothing to worry about.)
- In the OUTLINER, **click** on the *spine_cluster3handle* cluster; then holding down the **ctrl key**, **click** on the *spine_cluster4handle* cluster and the *hip_ctrl*. Press the **(p)** key to parent them. (This makes the *spine_cluster3handle* and *spine_cluster4handle* child to the *hip_ctrl*.)



Parenting the clusters to the the *spine_shoulder_ctrl* and *hip_ctrl*.

- (When *spine_ikHandle* is selected, notice that there is an attribute named *Twist*. If you **click** on the word *Twist* in the channel box, place your cursor over the PERSPECTIVE view panel, and **MMB click** and drag left to right, you will notice that the joint rotation is tapered down to the beginning of the IK chain at the *IK_spine1* joint. We can control this *Twist* attribute interactively with the *Advanced Twist* options in the attribute editor to provide natural left to right rotation of the spine when the shoulders rotate.)



Notice the wireframe of the skinned character on the left as compared to the character on the right. The character on the left is simply rotating the top of the spine, while the character on the right shows how the twist attribute creates natural rotations down the spine when the character turns from right to left.

- **Select** the *spine_ikHandle* and press **(ctrl+a)** to open attribute editor.
- **Scroll** down to the *IK Solver Attributes* and **click** on the arrow **(>)** to open the section.
- **Scroll** down to the *Advanced Twist Controls* and **click** on the arrow **(>)** to open the section.
- Under *Advanced Twist Controls* section, put a **check** next to "Enable Twist Controls" and **set** the following:

  World Up Type: choose "Object Rotation Up (Start/End)".

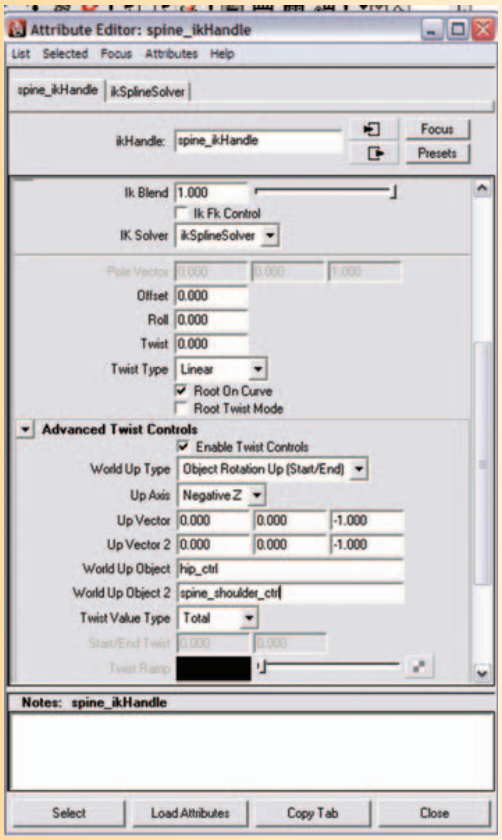
  Up Axis: choose "Negative Z".

  Up Vector: type **"0" "0"** "–1".

  Up Vector 2: type **"0" "0"** "–1".

  World Up Object: type "hip_ctrl".

  World Up Object 2: type "spine_shoulder_ctrl".

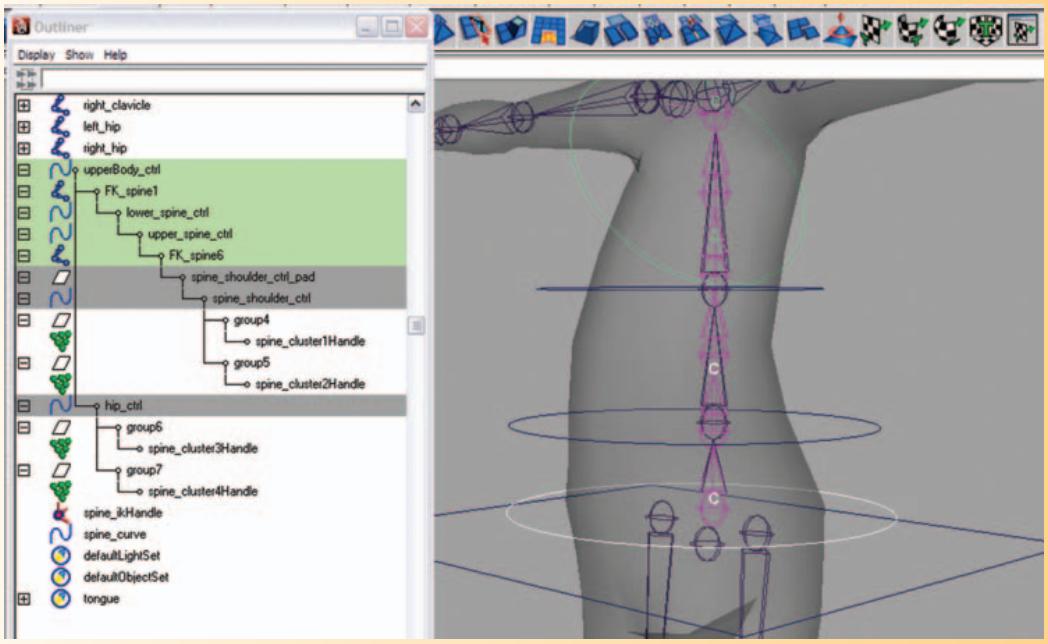


Setting the Advanced Twist Controls.

- **Change** the rotation order for the *hip_ctrl* by doing the following:
  - i. With the *hip_ctrl* selected, open the attribute editor by pressing **(ctrl+a)**.
  - ii. **Select** the *hip_ctrl* tab.
  - iii. Under *Transform Attributes* **set** the following:
    1. rotate order: choose **"ZXY"**.
- **Change** the rotation order for the *spine_shoulder_ctrl* by doing the following:
  - i. With the *spine_spine_shoulder_ctrl* selected, open the attribute editor by pressing **(ctrl+a)**.
  - ii. **Select** the *spine_spine_shoulder_ctrl* tab.
  - iii. Under *Transform Attributes* set the following:
    1. Rotate order: choose **"ZXY"**.

18. Integrate the IK spine into the existing spine controls by doing the following:
    - In the OUTLINER, **select** the *upperBody_ctrl* and press **(shift+h)** to display it.
    - In the OUTLINER, **hold down** the **shift** key and **click** on the plus sign **(+)** next to the *upperBody_ctrl* to open the hierarchy and display the children.
    - **Select** the *spine_shoulder_ctrl* and press **(ctrl+g)** to group the *spine_shoulder_ctrl* node. (We must group the *spine_shoulder_ctrl* to add a buffer between the joint and the controller as discussed earlier in this chapter.)

- In the OUTLINER, **double-click** on the group node and **rename** it *spine_shoulder_ctrl_pad*.
- In the OUTLINER, **hold down** the **MMB**, **click** on the *spine_shoulder_ctrl_pad* and **drag** it onto the *FK_spine6* joint. (This makes the *spine_shoulder_ctrl_pad* child to the *FK_spine6* joint.)
- In the OUTLINER, **hold down** the **MMB**, **click** on the *hip_ctrl* and **drag** it onto the *upperBody_ctrl*. (No group is necessary when the parent is a NURBS curve.)



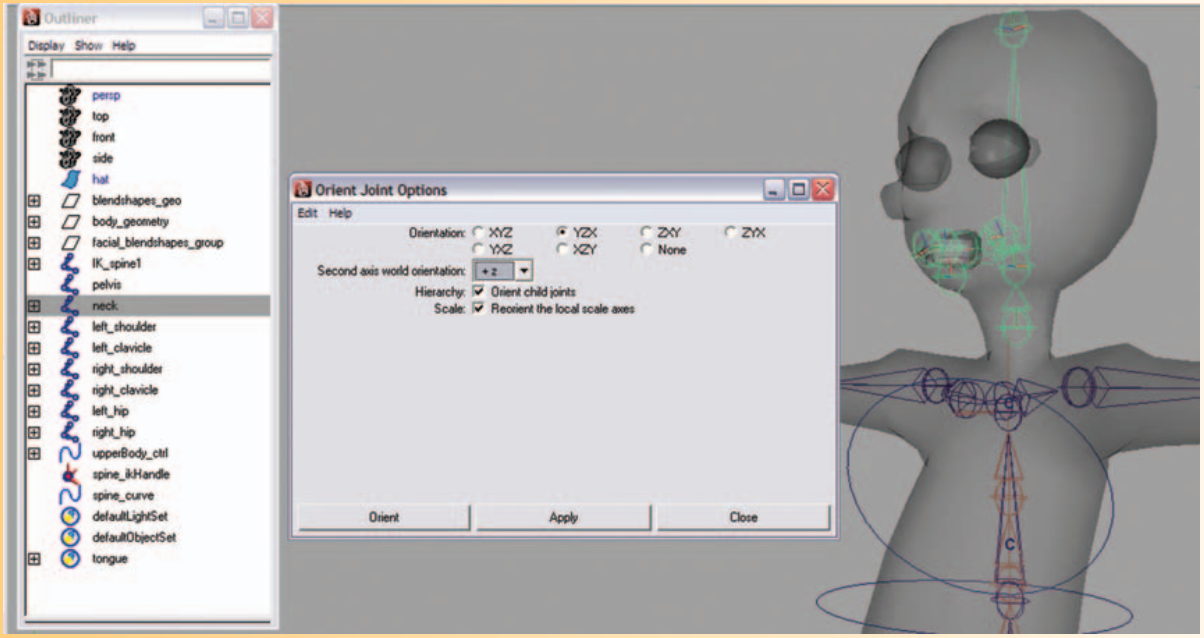
Integrating the IK spine into the FK spine.

19. **Save** your scene file. Name your scene *06_asgn01.ma*.

Assignment 6.2: Creating a Control System for the Neck and Head Skeleton

1. **Open** Maya and set your project.
   a. From your computer's desktop, go to **[Start > Programs]** and **select** Maya.
   b. Once Maya is open go to **[File > Project > Set...]** and browse to your project folder then **click "OK"**.
2. **Open** your last saved file: Go to **[File > Open]** and **select** *06_asgn01.ma*.
3. Continue working in X-ray mode.
4. Make sure that your geometry layer is set to **R** for reference so that you are unable to **select** the geometry by mistake when working.
5. To make selection easier, open your OUTLINER by going to **[Windows > Outliner]**.

6. First, we must reorient the neck for FK control by
   a. **Select** the *neck* joint and go to **[Skeleton > Orient Joint – option box]**. Set the following:
      i. Orientation: choose "**YZX**".
      ii. Second axis world orientation: **choose +Z**.
      (This will ensure that the joint orientation aligns with the world axis.)
      iii. **Click orient**.



Reorienting the *neck* joint chain.

   b. Go to **[Create > NURBS Primitives > Circle]**.
   c. In the MEL command line, **type** the MEL script below:

      parent -add -shape nurbsCircleShape1 neck;

      (This command gives a NURBS shape to the joint, making it easier to **select** during animation. It will appear that there are two NURBS circles in the scene.)
   d. **Resize** the circle to fit around your character's neck geometry by doing the following:
      i. Press the **(F8)** key.
      ii. **Choose** the "**select point components**" button in the Status Line.
      iii. Using the scale tool by pressing **(r)**, **click** and **drag** around the points of the circle and **scale** them larger so that it extends beyond the character's neck

to make it easier to **select** when animating. (This affects both circles visible.)

e. Press the **(F8)** key to go back into object mode.

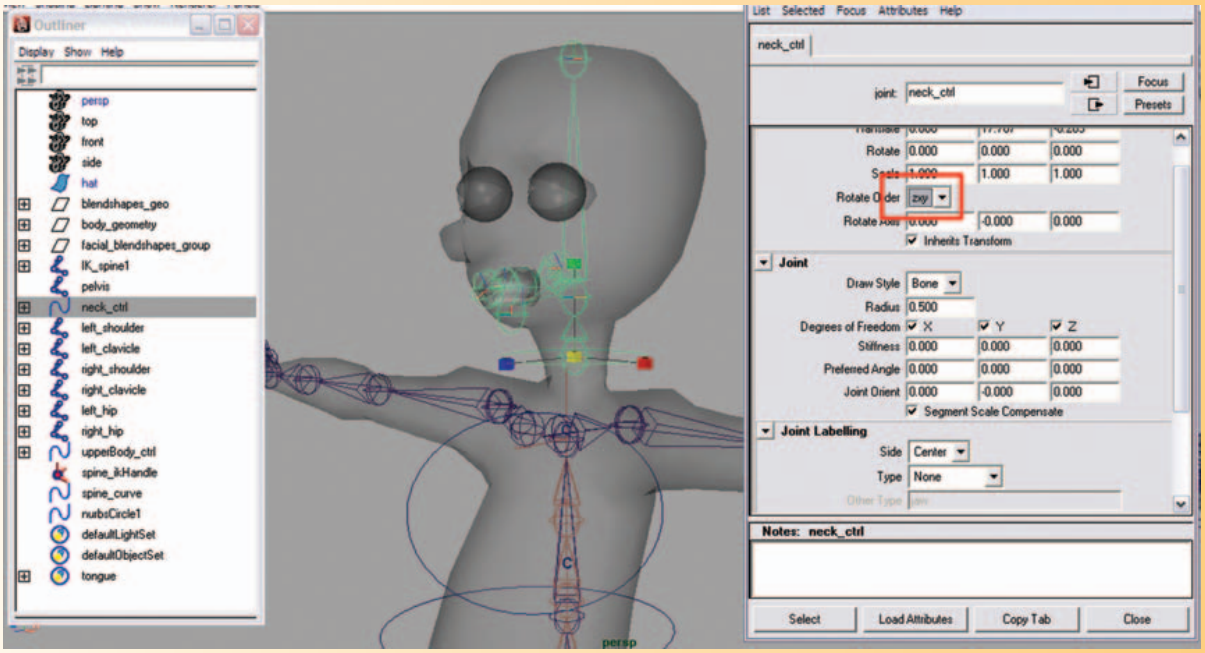
f. **Select** the *neck* joint, and in the channel box, **rename** *neck* to *neck_ctrl*.

g. **Change** the rotation order for the *neck_ctrl* by doing the following:

   i. With the *neck_ctrl* selected, **open** the attribute editor by pressing **(ctrl+a)**.

   ii. Under *Transform Attributes* **set** the following:

      1. Rotate order: choose "**ZXY**".



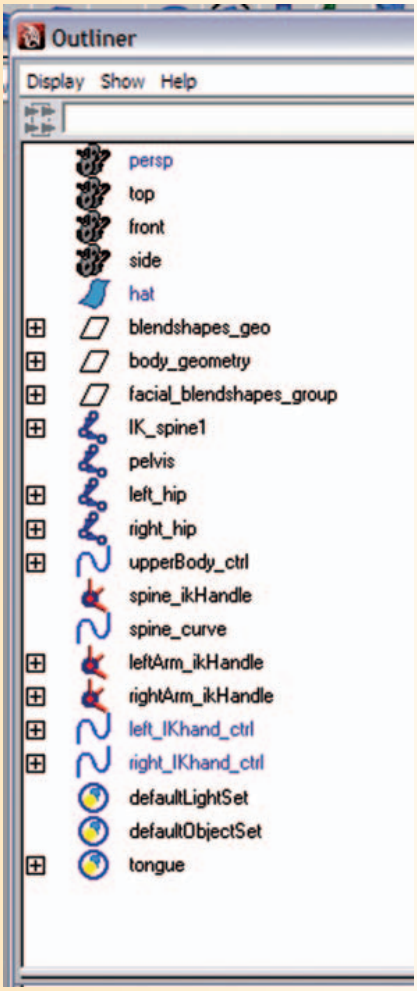
Adding a *nurbsCircleShape1* to the *neck* joint.

h. In the OUTLINER, go to **[Display]** and make sure there is a check mark next to *shapes*. If not, **click** on the word *shapes*.

i. In the OUTLINER, **hold down the shift key** and **click** on the plus sign **(+)** next to the *neck_ctrl* to open the hierarchy and display the children.

j. **Double-click** on *nurbsCircleShape1* and **rename** it *neckShape*. (We must rename the *nurbsCircleShape1* so that Maya does not get confused if we create more NURBS circles later.)

k. In the OUTLINER, **select** nurbsCircle1 and hit the **delete** key. (We no longer need the NURBS curves – we only needed the shape node.)



Finding the *nurbsCircleShape1* in the OUTLINER.

7. **Repeat** this process for the head joint by doing the following:
   a. Go to **[Create > NURBS Primitives > Circle]**.
   b. In the MEL command line, type the MEL script below:

      parent -add -shape nurbsCircleShape1 head;

   c. **Resize** the circle to fit around your character's head geometry by doing the following:
      i. Press the **(F8)** key to go into component mode.
      ii. **Choose** the "**select point components**" button in the Status Line.
      iii. Using the scale tool by pressing **(r)**, **click** and **drag** around the points of the new circle and **scale** them larger so that it extends beyond the character's head to make it easier to **select** when animating. (Some students like using the move tool by pressing **(w)** and moving the points up around the forehead area, like a head band, because the area around the neck is starting to get a bit congested with controllers – or you could rotate the points into a different direction, like in the image below.)

d. Press the **(F8)** key to go back into object mode.

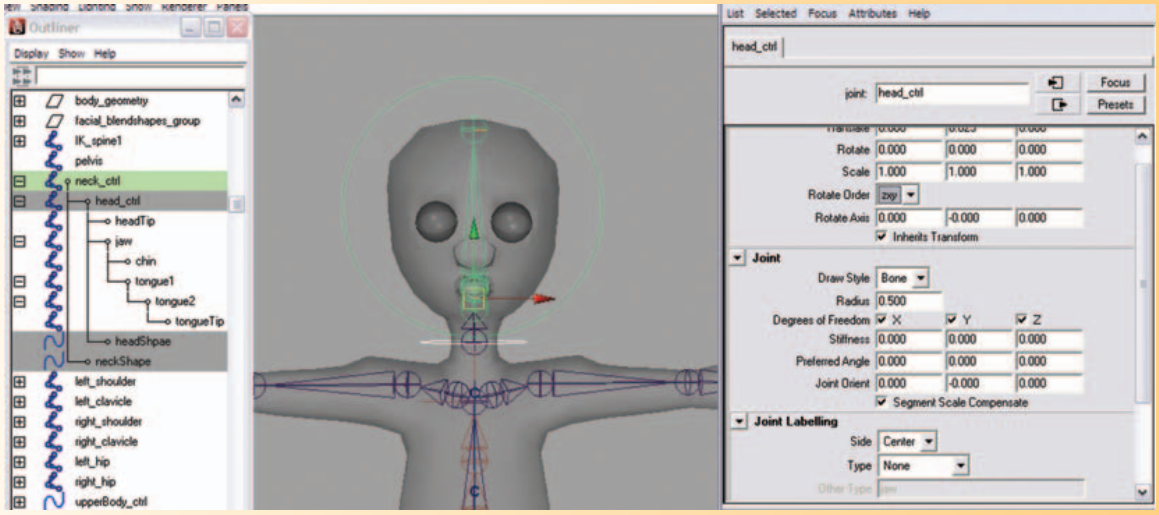
e. **Select** the *head* joint, and in the channel box, **rename** *head* to *head_ctrl*.

f. **Change** the rotation order for the *head_ctrl* by doing the following:

i. With the *head_ctrl* selected, open the attribute editor by pressing **(ctrl+a)**.

ii. Under *Transform Attributes* **set** the following:

1. Rotate order: choose "**ZXY**".



Adding a *nurbsCircleShape1* to the *head* joint.

g. In the OUTLINER, **select** nurbsCircle1 and hit the **delete** key.

h. **Double-click** on *nurbsCircleShape1* and **rename** it *headShape*.

8. Integrate the neck and head into the existing spine controls by doing the following:

a. In the PERSPECTIVE window, **select** the *neck_ctrl*, hold down the **(shift)** key and **click** the *shoulder_spine_ctrl*, and press **(p)** to parent.

9. Save your scene file. Name your scene *06_asgn02.ma*.

Assignment 6.3: Creating a Control System for the Clavicle

1. **Open** Maya and **set** your project.

a. From your computer's desktop, go to **[Start > Programs]** and **select** Maya.

b. Once Maya is open go to **[File > Project > Set...]** and browse to your project folder then **click "OK"**.

2. Open your last saved file: Go to **[File > Open]** and **select** *06_asgn02.ma*.

3. Continue working in X-ray mode.

4. Make sure that your geometry layer is set to **R** for reference so that you are unable to **select** the geometry by mistake when working.

5. To make selection easier, open your OUTLINER by going to **[Windows > Outliner]**.

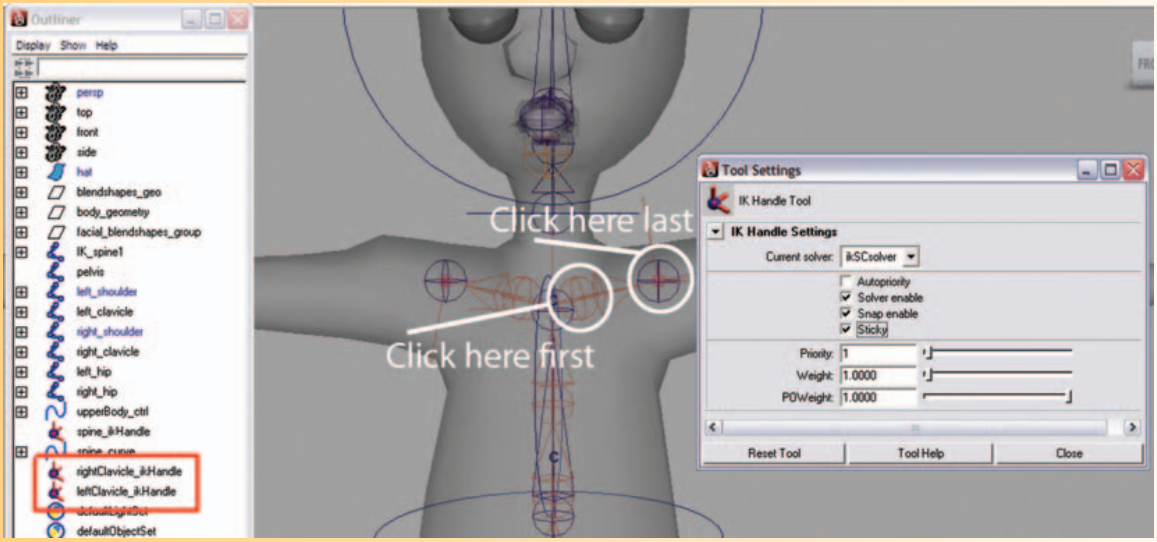
6. In the OUTLINER, **select** the *left_shoulder* joint chain and the *right_shoulder* joint chain – then press **(ctrl+h)** to hide the chain so that it is not in the way as we set up the arm.
7. **Create** the IK clavicle by doing the following:
   a. Go to **[Skeleton > IK Handle Tool – option box]** and set the following:
      i. **Click** "**reset tool**" then under *IK Handle Settings* **change** the following:

         Current solver: choose "**ikSCsolver**".

         Place a check mark in the box next to ***Sticky***.

         Then **click** "**close**".

   b. In the PERSPECTIVE window, **click** on the *left_clavicle* joint (to define the start of the IK joint chain) then **click** on the *left_claviclеTip* joint (to define the end of the chain. An IK handle appears at the end of the chain).
   c. In the OUTLINER, **double-click** on *ikHandle1* and **rename** it *leftClavicle_ikHandle*. (This chain will control the clavicle movement.)
   d. **Repeat** for the right clavicle.

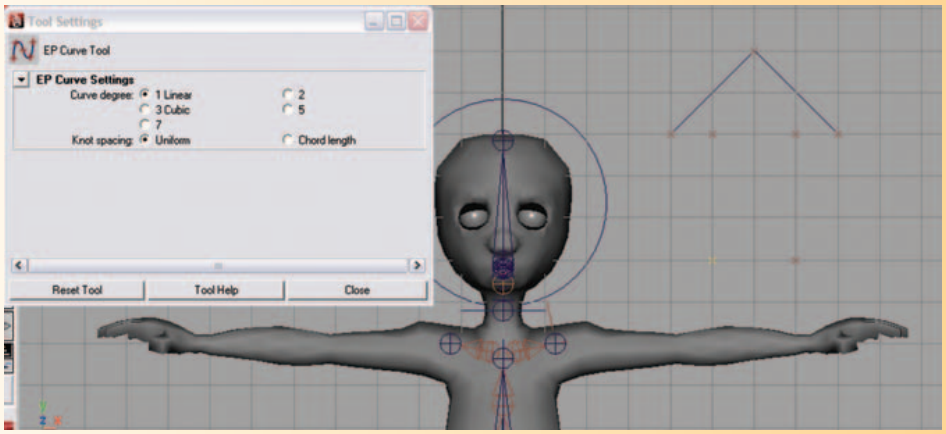


Creating the Clavicle_ikHandles.

8. **Create** a control system for the IK clavicle by doing the following:
   a. First **create** the controllers by doing the following:
      i. Go to **[Create > EP Curve Tool – option box]**
         1. Under *EP Curve Settings*, **change** the following:

            Curve degree: choose "**1 linear**".

      ii. In the FRONT view, use the grid snap tool by holding the **(x)** key and **click** to draw an arrow around the shoulder of your character. Hit "**enter**" when completed.



Drawing an arrow shaped curve using the EP curve tool.

iii. In the channel box, **rename** the arrow *left_clavicle_ctrl*.

iv. With the *left_clavicle_ctrl* selected go to **[Modify > Center Pivot]**.

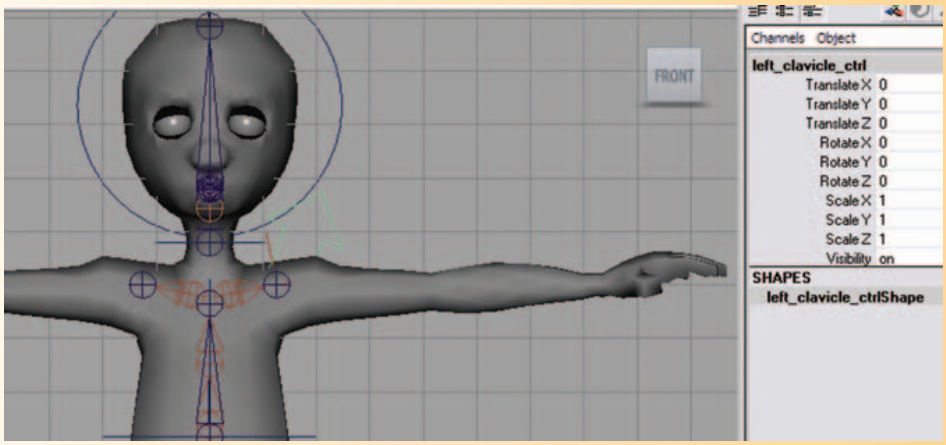
v. In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *left_claviclеTip* joint, and **click** the **MMB** and **drag** it slightly to snap the *left_clavicle_ctrl* into place.

vi. **Move and scale** the arrow by doing the following:

1. Press the **(F8)** key.
2. Choose the "**select** point components" button in the Status Line.

vii. Using the scale tool by pressing **(r)**, **click** and drag around the points of the arrow and scale them. This control should be scaled large enough that it is ABOVE the character's shoulder to make it easy to **select** when animating. Use the rotate tool by pressing **(e)** and rotate the arrow slightly away from the neck.

viii. With the *left_clavicle_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)



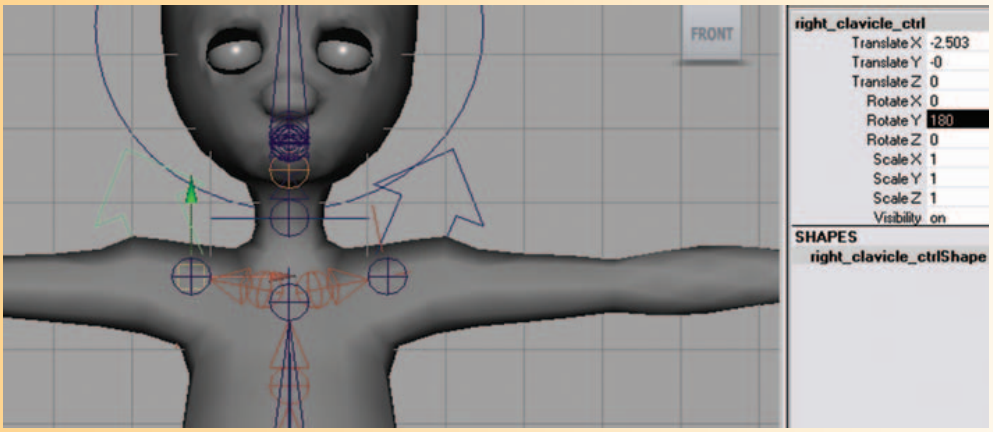
Positioning the *left_clavicle_ctrl*.

ix. **Duplicate** the *left_clavicle_ctrl* by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

x. In the OUTLINER, **double-click** on *left_clavicle_ctrl1* and **rename** it *right_clavicle_ctrl*.

xi. In PERSPECTIVE view, **select** the move tool by pressing **(w)**, **click** on the X axis **(red arrow)**, hold down the **(v)** key, position your cursor over the *right_claviculeTip* joint, and **click** the **MMB** and **drag** it slightly to snap the *right_clavicle_ctrl* into place.

xii. With the *right_clavicle_ctrl* selected **set** the following in the channel box: **RotateY:** *type* "**180**".



Duplicating the arrow and positioning the *left_clavicle_ctrl*.

xiii. Use the scale tool by pressing **(r)** and resize the arrow if necessary. With the *right_clavicle_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

xiv. The rotation order does not need to be changed on this controller, because rotations are not necessary for control.

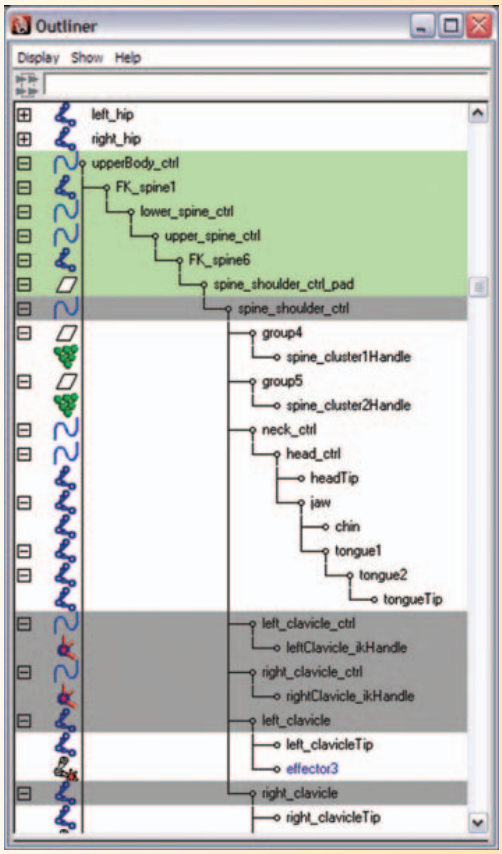
10. **Create** control between the controllers and the IK handles by doing the following:

    a. In the OUTLINER, **click** on the *leftClavicle_ikHandle* with the **MMB** and **drag** it onto the *left_clavicle_ctrl*. (This makes the *leftClavicle_ikHandle* child to the *left_clavicle_ctrl*.)

    b. In the OUTLINER, **click** on the *rightClavicle_ikHandle* with the **MMB** and **drag** it onto the *right_clavicle_ctrl*. (This makes the *rightClavicle_ikHandle* child to the *right_clavicle_ctrl*.)

11. Integrate the clavicles into the existing spine controls by doing the following:

    a. In the PERSPECTIVE window, **select** the *left_clavicle* joint, hold down the **(shift)** key and **click** the *right_clavicle* joint, the *left_clavicle_ctrl*, the *right_ clavicle_ctrl*, the *shoulder_spine_ctrl*, and then press **(p)** to parent.



The new OUTLINER hierarchy after integrating the clavicles into the existing spine controls.

12. In the OUTLINER, **select** the *left_shoulder* joint chain and *right_shoulder* joint chain and press **(shift+h)** to display them.
13. **Save** your scene file. Name your scene *06_asgn03.ma*.

Assignment 6.4: Creating a Control System for the Arm

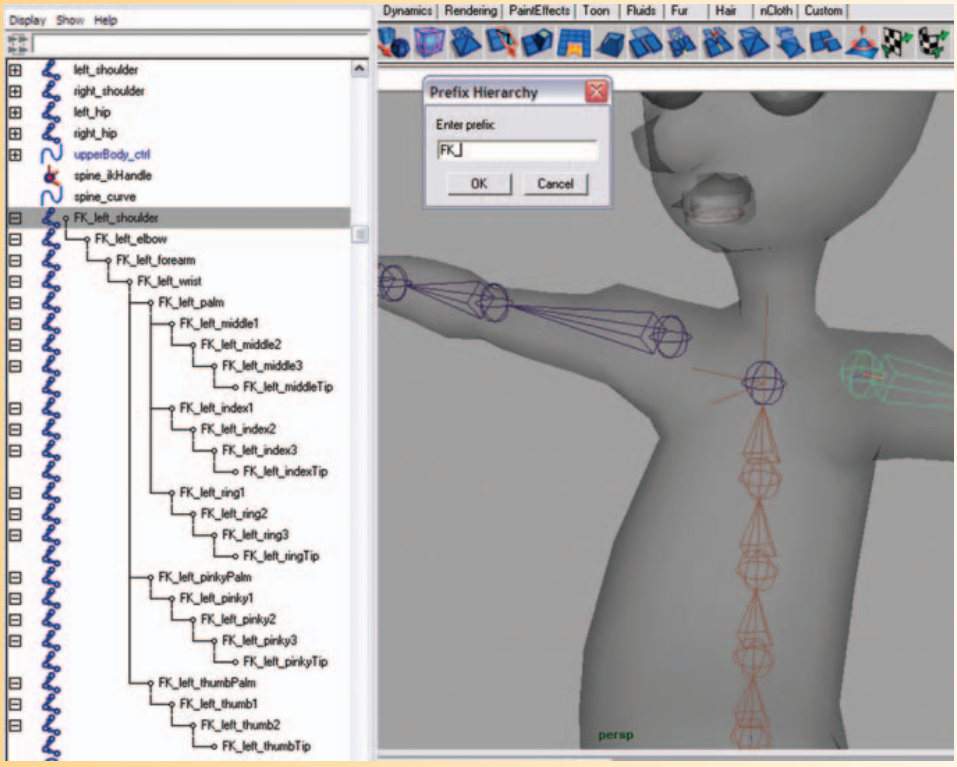
1. **Open** Maya and **set** your project.
   a. From your computer's desktop, go to **[Start > Programs]** and **select** Maya.
   b. Once Maya is open go to **[File > Project > Set...]** and browse to your project folder then **click** "**OK**".
2. Open your last saved file: Go to **[File > Open]** and **select** *06_asgn03.ma*.
3. Continue working in X-ray mode.
4. Make sure that your geometry layer is set to **R** for reference so that you are unable to **select** the geometry by mistake when working.
5. To make selection easier open your OUTLINER by going to **[Windows > Outliner]**.
6. In the OUTLINER, **select** the *upperBody_ctrl* and press **(ctrl+h)** to hide the chain, so that it is not in the way as we set up the arm.

We will be creating 3 joint chains for the arm. One chain will control the geometry, and two chains will have control systems – one for FK and one for IK. A switch will be

implemented for the chain that controls the geometry so that the animator can choose which control system it will follow.

7. **Create** an FK joint chain for the arm by doing the following:
   a. **Select** the *left_shoulder* joint.
   b. **Duplicate** it by going to **[Edit > Duplicate]** or press **(ctrl+d)**.
8. **Add** FK_ prefix to duplicated chain by **select**ing **[Modify > Prefix Hierarchy Names...]** and set the following:
   a. **Enter** prefix: "**FK_**".
   b. **Click OK**.

   (The duplicated joint chain now begins with the *FK_left_shoulder1*.)
9. In the OUTLINER, **double-click** on *FK_left_shoulder1* and **remove** the 1, renaming it to *FK_left_shoulder*.
10. In the OUTLINER, **hold down the shift key** and **click** on the plus sign **(+)** next to the *FK_left_shoulder* to open the hierarchy and display the children.



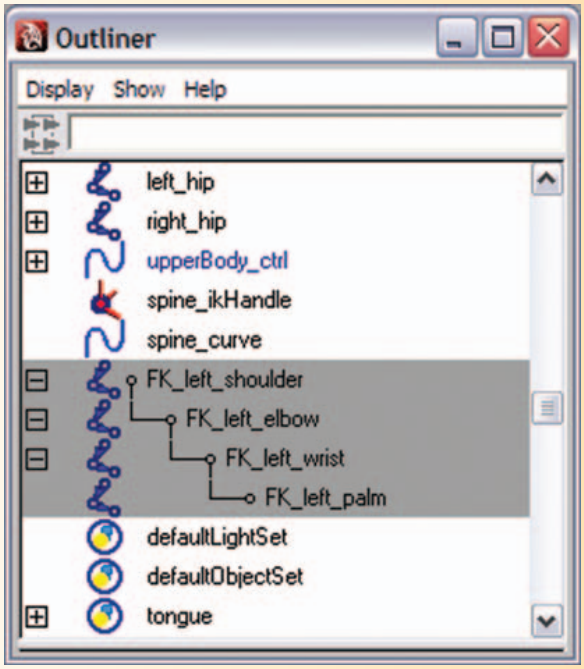
The new FK_left_shoulder hierarchy as seen in the OUTLINER.

11. For the FK chain, we will not need the *FK_left_forearm* joint. We can remove the extra joint from the chain by doing the following:
   a. **Select** the *FK_left_shoulder* joint.
   b. Go to **[Skeleton > Remove Joint]**.

12. In the OUTLINER, **hold down the shift key** and **click** on the plus sign **(+)** next to the *FK_left_wrist* to open the hierarchy and display the children.
13. For the FK chain, we will not need the *finger* joints. We can remove the extra joints from the chain by doing the following in the OUTLINER:
    a. **Select** the *FK_left_middle1* joint and hit the **(delete)** key on the keyboard.
    b. **Select** the *FK_left_ring1* joint and hit the **(delete)** key on the keyboard.
    c. **Select** the *FK_left_index1* joint and hit the **(delete)** key on the keyboard.
    d. **Select** the *FK_left_pinkyPalm* joint and hit the **(delete)** key on the keyboard.
    e. **Select** the *FK_left_thumbPalm* joint and hit the **(delete)** key on the keyboard.

    Your joint chain should be left with only four joints:

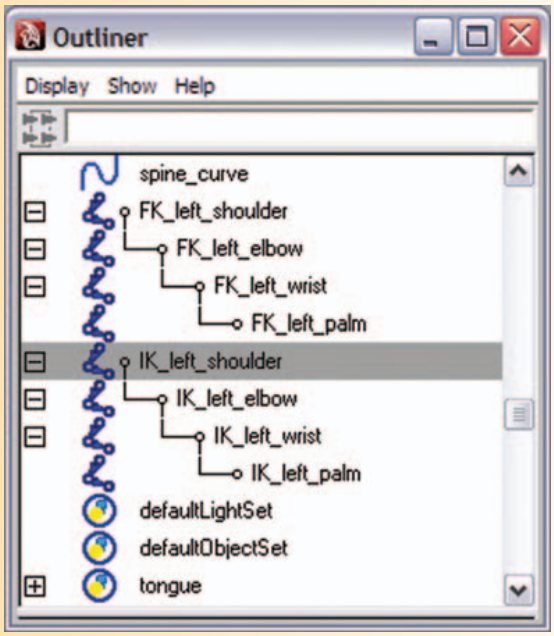
    FK_left_shoulder, *FK_left_elbow*, *FK_left_wrist*, and *FK_left_palm*.



The FK_left_shoulder hierarchy after removing the unnecessary joints.

14. **Create** an IK joint chain for the arm by doing the following:
    a. **Select** the *FK_left_shoulder* joint.
    b. **Duplicate** it by going to **[Edit > Duplicate]** or press **(ctrl+d)**.
15. **Select** the *FK_left_shoulder1* joint and **rename** the hierarchy by going to **[Modify > Search and Replace Names...]** and set the following:
    a. Search for: "**FK**".
    b. Replace with: "**IK**".

    (The duplicated joint chain now begins with the *IK_left_shoulder1*.)
16. In the OUTLINER, **double-click** on *IK_left_shoulder1* and **remove** the 1, renaming it to *IK_left_shoulder*.



The new IK_left_shoulder hierarchy as seen in the OUTLINER.

17. At this time, we can mirror this chain for the right side by doing the following:

    a. **Select** the *IK_left_shoulder* joint, then go to **[Skeleton > Mirror Joint – option box]** and **enter** the following:

    Mirror Across: choose "**YZ**" axis.

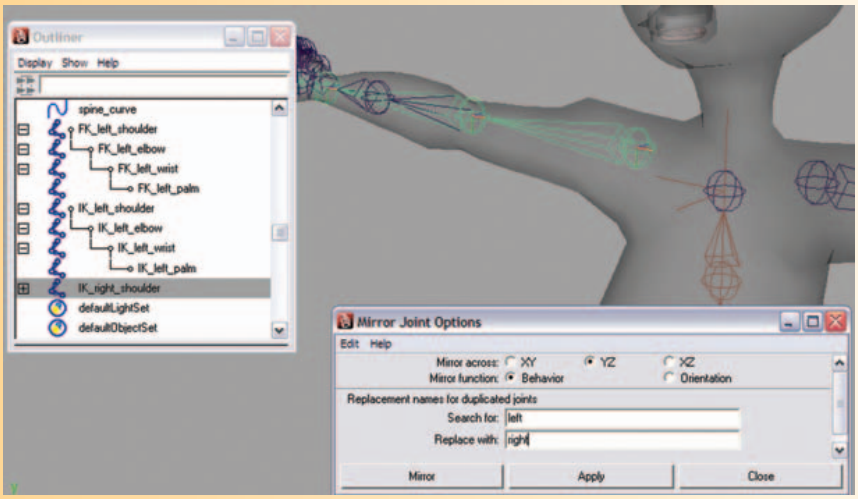
    Mirror Function: choose "**behavior**".

    Replacement names for duplicated joints:

    Search for: **enter** "**left**".

    Replace with: **enter** "right".

    Then **click** "**mirror**" to execute the command. (We will not mirror the FK arm at this time. Instead, we will now create the FK arm controls first and then mirror, saving time, and some repetition.)



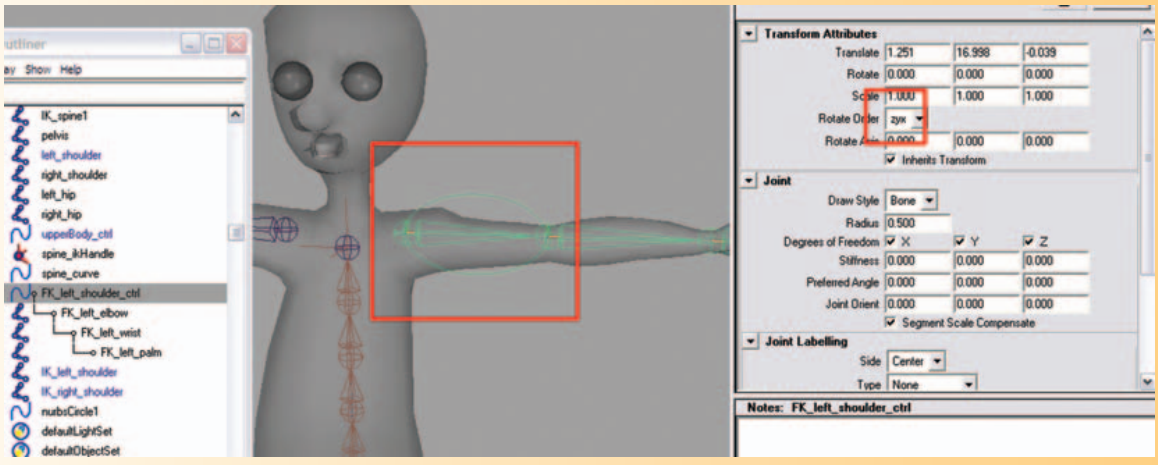
Mirroring the *IK_left_shoulder* joint for the right arm.

Remember, the FK arm should be created first so that the correct rotational orders can be set. It is very helpful to HIDE the joint chains that you don't need so that they do not get in the way while creating the FK controls.

18. In the OUTLINER, **select** the *IK_left_shoulder* joint chain and the *IK_right_shoulder* joint chain – then press **(ctrl+h)** to hide them.
19. In the OUTLINER, **select** the *left_shoulder* joint chain and the *right_shoulder* joint chain – then press **(ctrl+h)** to hide them.
20. **Create** controls for the FK arm by doing the following:
    a. Go to **[Create > NURBS Primitives > Circle]**.
    b. In the MEL command line, **type** the MEL script below:

       parent -add -shape nurbsCircleShape1 FK_left_shoulder;

    c. **Resize** the circle to fit around your character's upper arm geometry by doing the following:
       i. Press the **(F8)** key.
       ii. Choose the "**select point components**" button in the Status Line.
       iii. Using the scale tool by pressing **(r)**, the move tool by pressing **(w)**, and the rotate tool by pressing **(e)**, **click** and **drag** around the points of the circle and **scale** them larger so that it extends beyond the character's arm. **Position** it over the upper arm, and **rotate** to face forward to make it easier to **select** when animating.
    d. Press the **(F8)** key to go back into object mode.
    e. **Select** the *FK_left_shoulder* joint, and in the channel box, **rename** *FK_left_shoulder* to *FK_left_shoulder_ctrl*.
    f. **Change** the rotation order for the *FK_left_shoulder_ctrl* by doing the following:
       i. With the *FK_left_shoulder_ctrl* selected, open the attribute editor by pressing **(ctrl+a)**.
       ii. Under *Transform Attributes* set the following:
           1. Rotate order: choose "XZY".

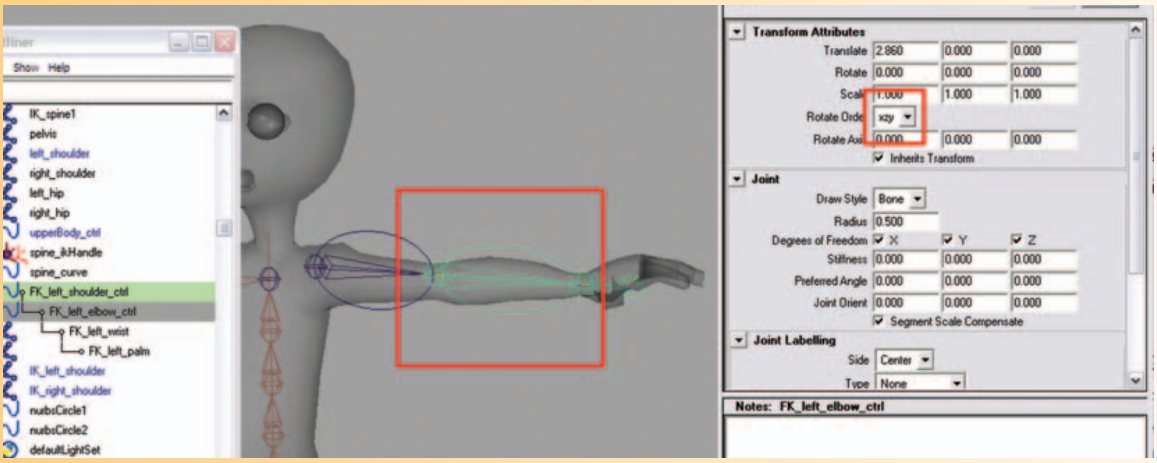


Adding a *nurbsCircleShape1* to the *FK_left_shoulder* joint.

21. Repeat this process for the elbow joint by doing the following:
    a. Go to **[Create > NURBS Primitives > Circle]**.
    b. In the MEL command line, type the MEL script below:

       parent -add -shape nurbsCircleShape2 FK_left_elbow;

    c. Resize the circle to fit around your character's forearm geometry by doing the following:
        i. Press the **(F8)** key to go into component mode.
        ii. Choose the "**select** point components" button in the Status Line.
        iii. Using the scale tool by pressing **(r)**, the move tool by pressing **(w)**, and the rotate tool by pressing **(e)**, **click** and drag around the points of the circle and scale them larger so that it extends beyond the character's forearm, position it over the upper arm, and rotate to face forward to make it easier to **select** when animating.
    d. Press the **(F8)** key to go back into object mode.
    e. **Select** the *FK_left_elbow* joint, and in the channel box, rename *FK_left_elbow* to *FK_left_elbow_ctrl*.
    f. Change the rotation order for the *FK_left_elbow_ctrl* by doing the following:
        i. With the *FK_left_elbow_ctrl* selected, open the attribute editor by pressing **(ctrl+a)**.
        ii. Under *Transform Attributes* set the following:
            1. Rotate order: choose "XZY".



Adding a *nurbsCircleShape2* to the *FK_left_elbow* joint.

22. Repeat this process for the wrist joint by doing the following:
    a. Go to **[Create > NURBS Primitives > Circle]**.
    b. In the MEL command line, type the MEL script below:

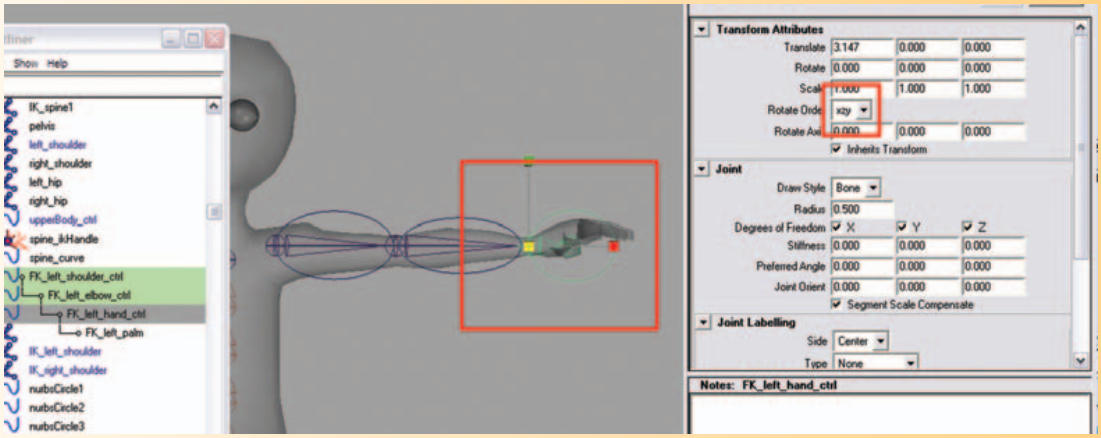
       parent -add -shape nurbsCircleShape3 FK_left_wrist;

c. Resize the circle to fit around your character's hand geometry by doing the following:
   i. Press the **(F8)** key to go into component mode.
   ii. Choose the "**select** point components" button in the Status Line.
   iii. Using the scale tool by pressing **(r)**, the move tool by pressing **(w)**, and the rotate tool by pressing **(e)**, **click** and drag around the points of the circle and scale them larger so that it extends beyond the character's hand, position it over the upper arm, and rotate to face forward to make it easier to **select** when animating.

d. Press the **(F8)** key to go back into object mode.

e. **Select** the *FK_left_wrist* joint, and in the channel box, rename *FK_left_wrist* to *FK_left_hand_ctrl*.

f. Change the rotation order for the *FK_left_hand_ctrl* by doing the following:
   i. With the *FK_left_hand_ctrl* selected, open the attribute editor by pressing **(ctrl+a)**.
   ii. Under *Transform Attributes* set the following:
      1. Rotate order: choose "ZYX".



Adding a *nurbsCircleShape3* to the *FK_left_wrist* joint.

g. In the OUTLINER, **select** nurbsCircle1, nurbsCircle2, and nurbsCircle3 and hit the delete key. (We no longer need the NURBS curves – we only needed their shape nodes.)

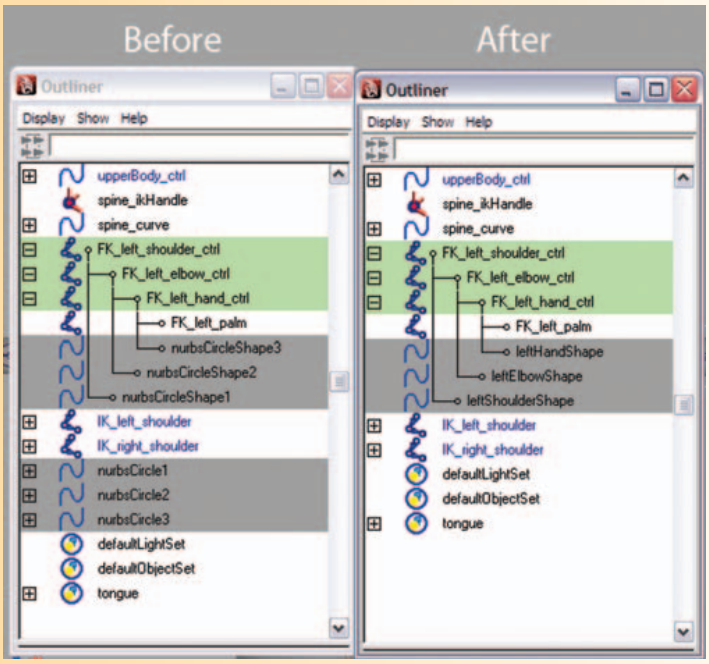
h. In the OUTLINER, go to **[Display]** and make sure there is a check mark next to *shapes*. If not, **click** on the word *shapes*.

i. In the OUTLINER, hold down the shift key and **click** on the plus sign **(+)** next to the *FK_left_spine_shoulder_ctrl* to open the hierarchy and display the children.

j. Double-**click** on *nurbsCircleShape1* and **rename** it *leftShoulderShape* (we must rename the *nurbsCircleShape1* so that Maya does not get confused if we create more NURBS circles later.

k. Double-**click** on *nurbsCircleShape2* and **rename** it *leftElbowShape*.

l. Double-**click** on *nurbsCircleShape3* and **rename** it *leftHandShape*.



Renaming *nurbsCircleShape1, nurbsCircleShape2, nurbsCircleShape3* in the OUTLINER, and deleting the NURBS circle curves.

23. At this time, we can mirror this chain for the right side by doing the following:

    a. **Select** the *FK_left_shoulder_ctrl* joint, then go to **[Skeleton > Mirror Joint – option box]** and **enter** the following:

    Mirror Across: choose "YZ" axis.

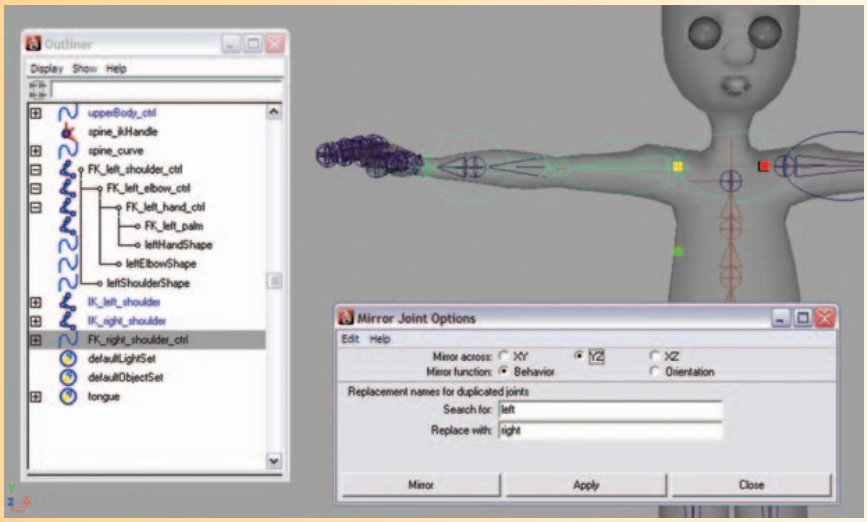
    Mirror Function: choose "behavior".

    Replacement names for duplicated joints:

    Search for: **enter** "left".

    Replace with: **enter** "right".

    Then **click** "mirror" to execute the command.



Mirroring the *FK_left_shoulder_ctrl* for the right arm.

b. When the arm is mirrored, the NURBS shapes do not. We must move them into the correct area of the arm. To do this, **select** the *FK_right_hand_ctrl* and reposition the curve by doing the following:

   i. Press the **(F8)** key.

   ii. Choose the "**select** point components" button in the Status Line.

   iii. Using the move tool by pressing **(w)**, **click** and drag around the points of the circle and position them over the right hand.

   iv. In the channel box, rename.

c. **Select** the *FK_right_elbow_ctrl* and reposition the curve by doing the following:

   i. Press the **(F8)** key.

   ii. Choose the "**select** point components" button in the Status Line.

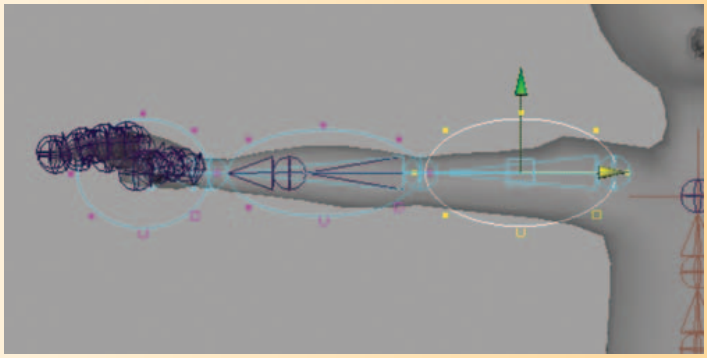
   iii. Using the move tool by pressing **(w)**, **click** and drag around the points of the circle and position them over the right forearm.

d. **Select** the *FK_right_shoulder_ctrl* and reposition the curve by doing the following:

   i. Press the **(F8)** key.

   ii. Choose the "**select** point components" button in the Status Line.

   iii. Using the move tool by pressing **(w)**, **click** and drag around the points of the circle and position them over the right upper arm.



Positioning the mirrored curves over the right arm.

24. Because there still might be problems with Gimbal lock, we can add one more control above the shoulder to add another level of control to position the FK arm.

   a. First create the controller by doing the following:

      i. Go to **[Create > NURBS Primitives > Circle]**.

      ii. In the channel box, rename the circle *left_FKarm_ctrl*.

      iii. In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *FK_left_shoulder* joint, and **click** the **MMB** and drag it slightly to snap the *left_FKarm_ctrl* into place.

iv. Use the scale tool by pressing **(r)** and resize the circle if necessary. (This control should be scaled large enough that it is OUTSIDE of the character's arm to make it easy to **select**.)

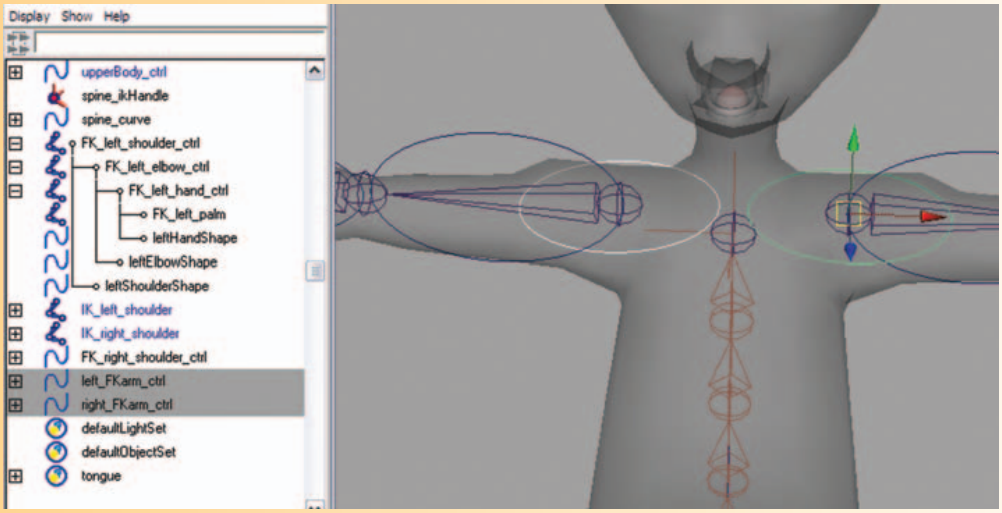
v. With the *left_FKarm_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

vi. Duplicate the *left_FKarm_ctrl* by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

vii. In the OUTLINER, **double-click** on *left_FKarm_ctrl1* and rename it *right_FKarm_ctrl*.

viii. In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *FK_right_shoulder* joint, and **click** the **MMB** and drag it slightly to snap the *right_FKarm_ctrl* into place.

ix. With the *right_FKarm_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)



Creating the left and right FKarm_ctrl to aid in the prevention of Gimbal lock.

b. In the OUTLINER, **click** on the *FK_left_shoulder_ctrl* with the **MMB** and drag it onto the *left_FKarm_ctrl*.

c. In the OUTLINER, **click** on the *FK_right_shoulder_ctrl* with the **MMB** and drag it onto the *right_FKarm_ctrl*.

25. Now that we are finished with the FK arm, we can hide it to work on the IK arm. In the OUTLINER, **select** the *left_FKarm_ctrl* and the *right_FKarm_ctrl* – then press **(ctrl+h)** to hide them.

26. In the OUTLINER, **select** the *IK_left_shoulder* joint chain and the *IK_right_shoulder* joint chain – then press **(shift+h)** to display the chain.

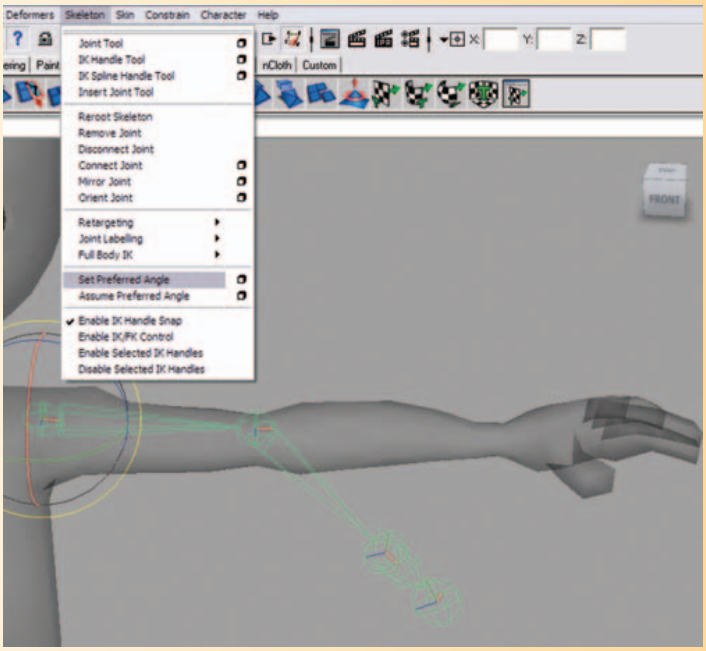
27. Create the left IK arm by doing the following:

    a. Set a preferred angle in the left arm by doing the following:

        i. **Select** the *IK_left_elbow* joint and in the channel box set the following: RotateY: type "−25".

        ii. **Select** the *IK_left_shoulder* joint, then go to **[Skeleton > Set Preferred Angle]**.

        iii. **Select** the *IK_left_elbow* joint and in the channel box set the following: RotateY: type "**0**". (MAKE SURE YOU DO THIS to make the elbow straight again.)



Setting a preferred angle on the left arm.

> **!** We must first set a **preferred angle** in the arm so that Maya knows which direction to bend the arm when we run the IK solver through the joints. Be sure to straighten the arm again afterward.

    b. Go to **[Skeleton > IK Handle Tool – option box]** and set the following:

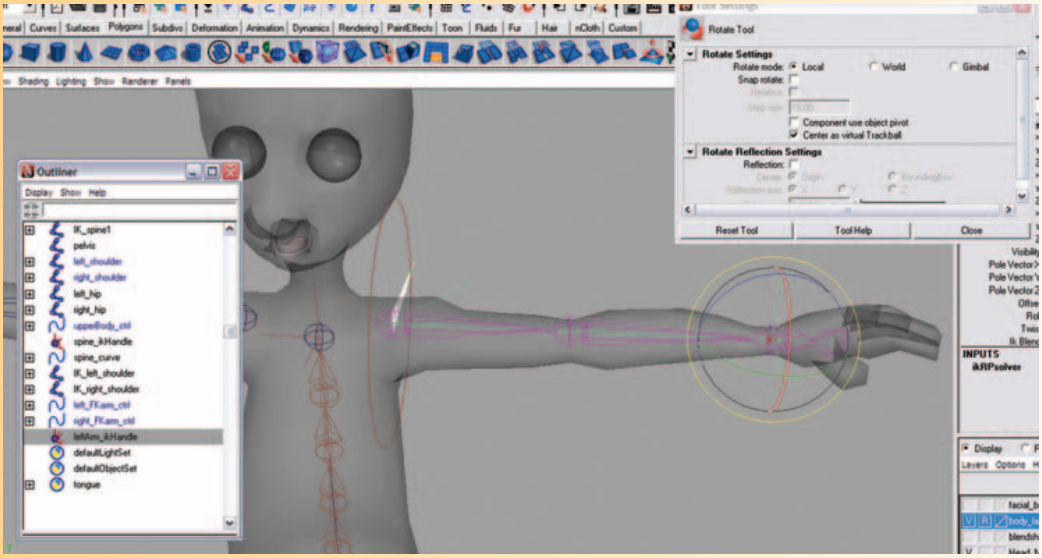
        i. **Click** "reset tool" then under *IK Handle Settings* change the following:

            Place a check mark in the box next to *Sticky*.

            Then **click** "close".

    c. In the PERSPECTIVE window, **click** on the *IK_ left_shoulder* joint (to define the start of the IK joint chain) then on the *IK_ left_wrist* joint. (To define the end of the chain. An IK handle appears at the end of the chain.)

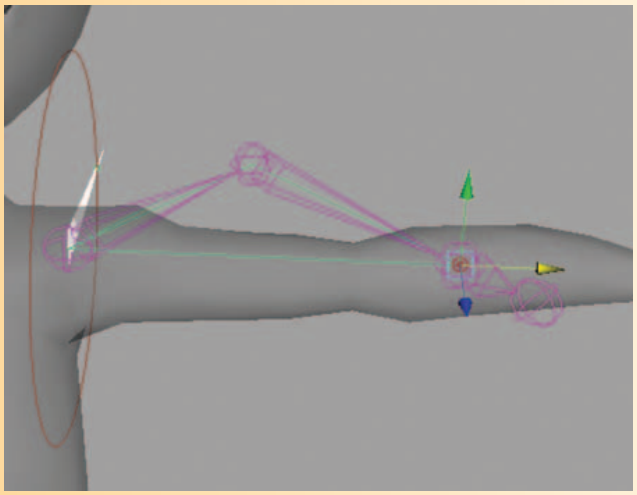
    d. In the OUTLINER, double-**click** on *ikHandle1* and rename it *leftArm_ikHandle*. (This chain will control the arm movement.)



Creating an RP IK solver in the left arm.

e. It is a great idea to check to make sure the IK handle moves correctly along the X axis. You can do this by doing the following:

f. **Select** the *leftArm_ikHandle* and move the IK handle along the X axis **(red arrow)** toward the body to confirm that the arm bends in the correct direction. Be sure to press the **(z)** key to undo the move.



Moving the leftArm _ikHandle along the X axis to make sure the arm bends correctly.

g. Go to **[Skeleton > IK Handle Tool – option box]** and set the following:

i. Under *IK Handle Settings* change the following:

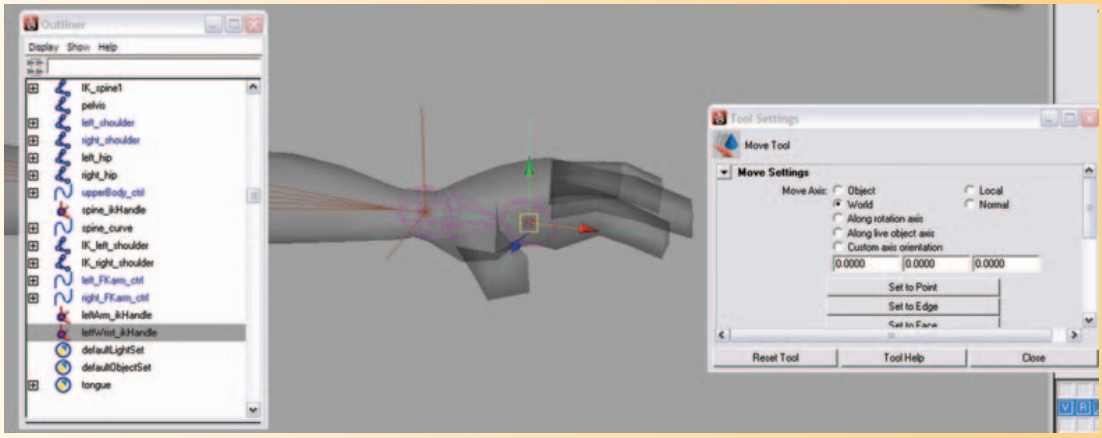
Current solver: choose "ikSCsolver".

Keep a check mark in the box next to *Sticky*.

Then **click** "close".

h. In the PERSPECTIVE window, **click** on the *IK_ left_wrist* joint (to define the start of the IK joint chain) then on the *IK_ left_palm* joint. (To define the end of the chain. An IK handle appears at the end of the chain.)

i. In the OUTLINER, double-**click** on *ikHandle1* and rename it *leftWrist_ikHandle*. (This chain will control the wrist movement.)

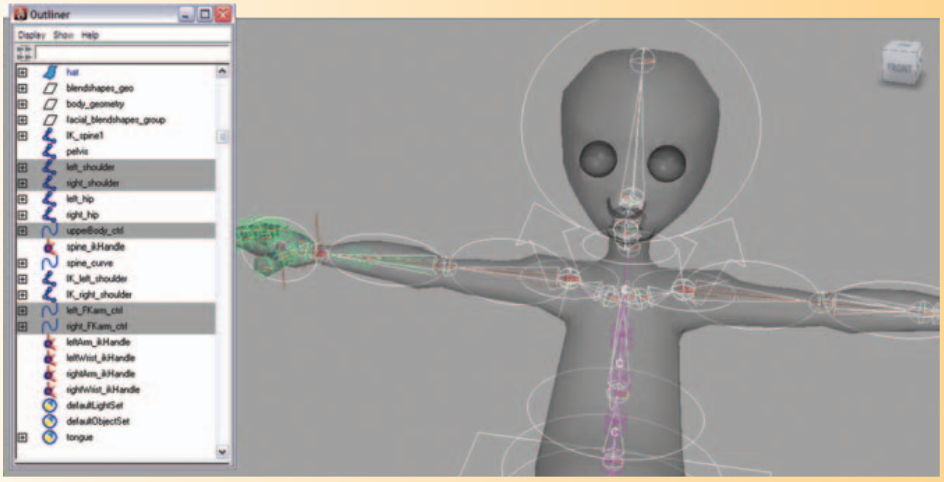


Creating a Single Chain IK solver in the left wrist.

> **!** Unfortunately, IK solvers do not mirror the way we would like them to behave, so we will need to repeat the steps for the right side. You can repeat the right arm when suggested to do so, or wait until the end to redo the right side all at once.

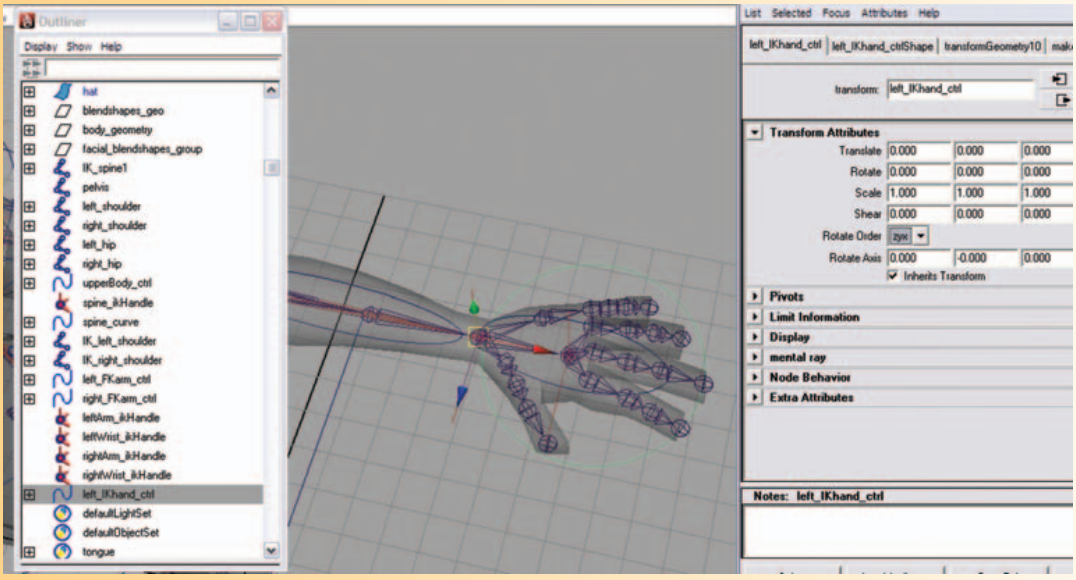
28. Repeat the above to create the right IK arm.

29. Now that we are finished with the IK arm, we make everything visible again. In the OUTLINER, **select** the *left_FKarm_ctrl*, hold down the **(ctrl)** key and **select** the *right_FKarm_ctrl*, the *left_shoulder* joint chain, the *right_shoulder* joint chain, and the *UpperBody_ctrl* – then press **(shift+h)** to display them.



Displaying the hidden arms in the OUTLINER.

30. Create a control system for the IK arm by doing the following:
    a. First create the controllers by doing the following:
        i. Go to **[Create > NURBS Primitives > Circle]**.
        ii. In the channel box, rename the circle *left_IKhand_ctrl*.
        iii. In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *IK_left_wrist* joint, and **click** the **MMB** and drag it slightly to snap the *left_IKhand_ctrl* into place.
        iv. Use the scale tool by pressing **(r)** and resize the circle if necessary. (This control should be scaled large enough that it is OUTSIDE of the character's arm to make it easy to **select**.)
        v. With the *left_IKhand_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)
        vi. **Select** the *left_IKhand_ctrl* and reposition the curve around the hand by doing the following:
            1. Press the **(F8)** key.
            2. Choose the "**select** point components" button in the Status Line.
            3. Using the move tool by pressing **(w)**, **click** and drag around the points of the circle and position them over the right wrist.
        vii. Change the rotation order for the *left_IKhand_ctrl* by doing the following:
            1. With the *left_IKhand_ctrl* selected, open the attribute editor by pressing **(ctrl+a)**.
            2. Under *Transform Attributes* set the following:
                a. Rotate order: choose "ZYX".



Creating and positioning the left_IKhand_ctrl to control the leftArm_ikHandle.

viii. Duplicate the *left_IKhand_ctrl* by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

ix. In the OUTLINER, **double-click** on *left_IKhand_ctrl1* and rename it *right_IKhand_ctrl*.

x. In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *IK_right_wrist* joint, **click** the **MMB** and drag it slightly to snap the *right_IKhand_ctrl* into place.

xi. With the *right_IKhand_ctrl* selected set the following in the channel box: RotateZ: type "180" (this will flip the controller over the right hand).

xii. With the *right_IKhand_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

xiii. Go to **[Create > NURBS Primitives > Circle]**.

xiv. In the channel box, rename the circle *left_IKelbow_ctrl*.

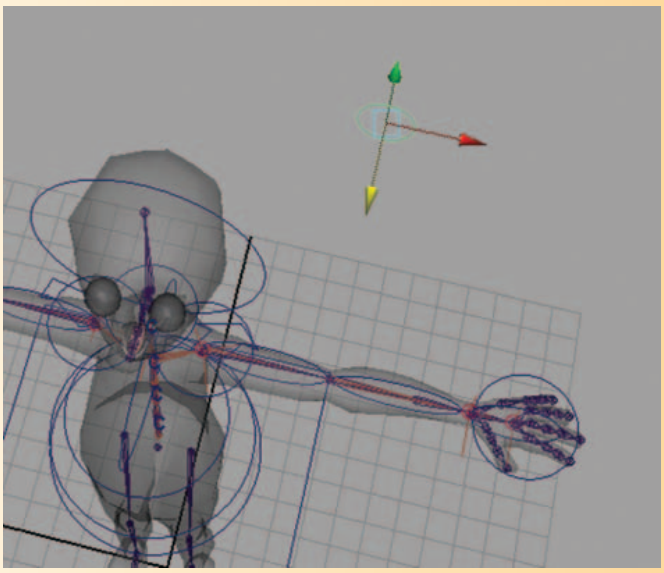
xv. In PERSPECTIVE view, with the *left_IKelbow_ctrl* selected, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *IK_left_elbow* joint, **click** the **MMB** and drag it slightly to snap the left_*IKelbow_ctrl* into place.

xvi. With the move tool, **click** on the Z axis **(blue arrow)** and move the controller arm distance behind the character.

xvii. With the *left_IKelbow_ctrl* selected set the following in the channel box: RotateX: type "90".

xviii. Use the scale tool by pressing **(r)** and resize the circle if necessary.

xix. With the *left_IKelbow_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)



Creating and positioning the left_IKelbow_ctrl to control the leftArm_ikHandle's pole vector.

xx. The rotation order does not need to be changed on this controller because rotations are not necessary for control.

xxi. Duplicate the *left_IKelbow_ctrl* by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

xxii. In the OUTLINER, **double-click** on *left_IKelbow_ctrl1* and rename it *right_IKelbow_ctrl*.

xxiii. In PERSPECTIVE view, **select** the move tool by pressing **(w)** and **click** on the X axis **(red arrow)**, hold down the **(v)** key, position your cursor over the *IK_right_elbow* joint, **click** the LMB (left mouse button) and drag it slightly to snap the *right_IKelbow_ctrl* into place. (By **select**ing the X axis first, the move is constrained to that axis only.)

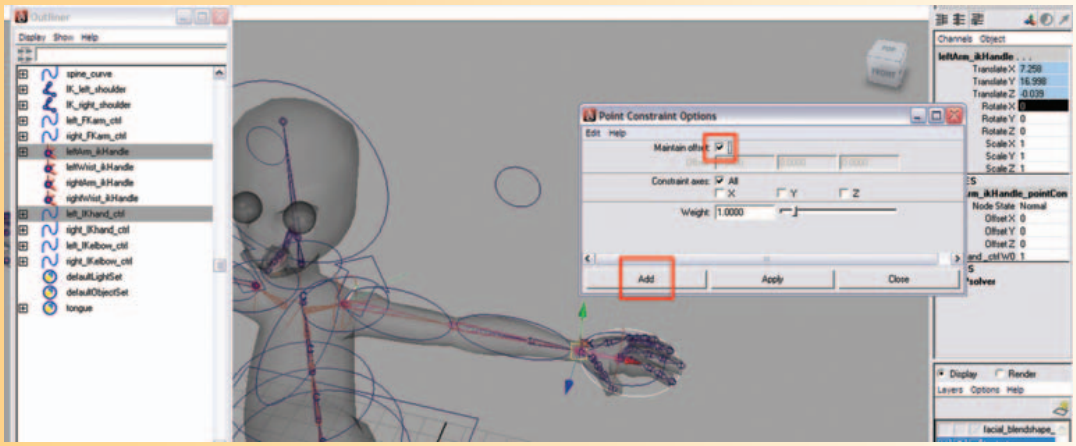
xxiv. With the *right_IKelbow_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

31. Create control between the controllers and the IK handles by doing the following:

a. In the OUTLINER, **click** on the *left_IKhand_ctrl* (the leader), hold down the **(ctrl)** key and **click** on the *leftArm_ikHandle* (the follower), then go to **[Constrain > Point – option box]** and set the following:

1. Go to **[Edit > Reset Settings]**.
2. Place a check mark in the box next to *Maintain Offset*.

Then **click** "**add**".

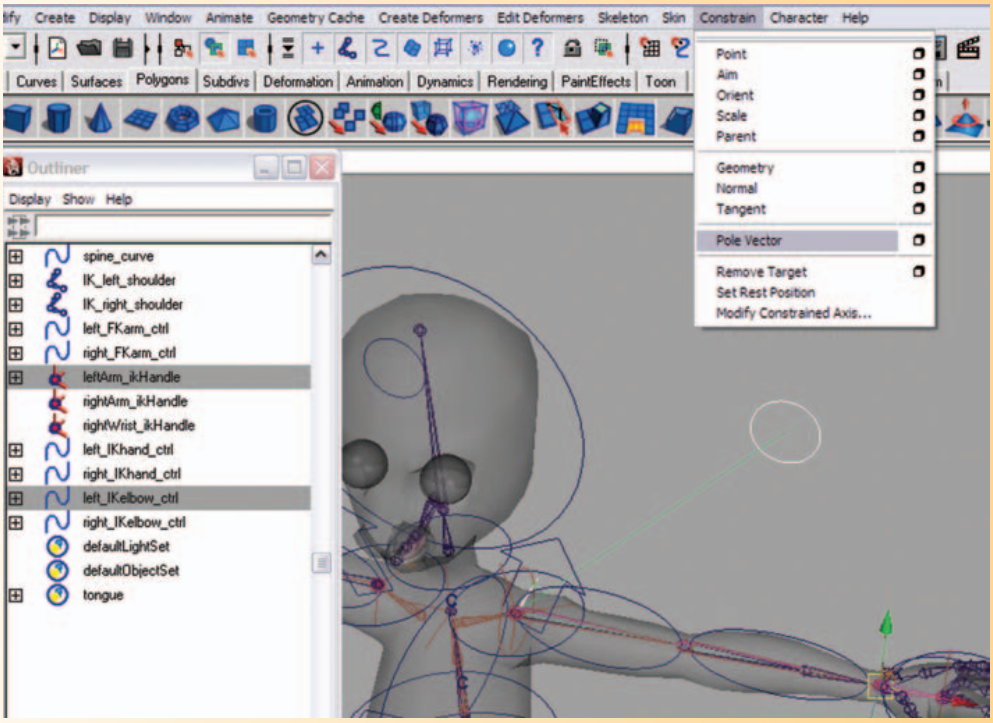


Creating a point constraint between the left_IKhand_ctrl (leader) and the leftArm_ikHandle (follower).

! This point constraint constrains the arm IK handle's translations to the controller so that when you move the controller, the IK handle follows it, BUT, when you rotate the controller, the IK handle does not. This is important in the arm setup because we don't want the entire arm to rotate when we rotate the hand.

b. In the OUTLINER, **click** on the *leftWrist_ikHandle* with the **MMB** and drag it onto the *left_IKhand_ctrl*. (This makes the *leftWrist_ikHandle* child to the *left_IKhand_ctrl* so that it moves and rotates with the *left_IKhand_ctrl*, which is desirable in this case so that the wrist rotates with the forearm.)

c. In the OUTLINER, **click** on the *left_IKelbow_ctrl* (the leader), hold down the **(ctrl)** key and **click** on the *leftArm_ikHandle* (the follower), then go to **[Constrain > Pole Vector]**.



Creating a pole vector constraint between the *left_IKelbow_ctrl* (leader) and the *leftArm_ikHandle* (follower).

d. Repeat the constraints and parenting for the right arm.

e. Integrate the elbow controls into the existing spine controls by doing the following:

i. In the PERSPECTIVE window, **select** the *left_IKelbow_ctrl*, shift **select** the *right_IKelbow_ctrl*, the *spine_shoulder_ctrl*, and press **(p)** on the keyboard to parent.

! You can now test out your arm controls. **Select** the left_IKhand_ctrl and move it around toward the body so that the elbow bends. You can also rotate this control to control the wrist. **Select** the left_IKelbow_ctrl and move it up and down to control the position of the elbow. Be sure to press the **(z)** key several times to undo the moves.

32. The next part will create an integrated arm where the joint chain that will eventually control the geometry has a switch to choose between the FK control arm and the IK control arm. Do the following:

a. Create a constraint system with two leaders, the FK control arm and the IK control arm. To set this up, do the following:

i. In the OUTLINER, **select** the *left_shoulder*, and while holding down the **(ctrl)** key, **select** the *left_FKarm_ctrl* and the *IK_left_shoulder*.

ii. Press **(ctrl+g)** to create a group of the three arms to make it easier to select.

iii. In the OUTLINER, **double-click** on *group1* and rename it *leftArm_grp*.

iv. In the OUTLINER, **select** the *right_shoulder*, and while holding down the **(ctrl)** key, **select** the *right_FKarm_ctrl* and the *IK_right_shoulder*.

v. Press **(ctrl+g)** to create a group of the three arms to make it easier to select.

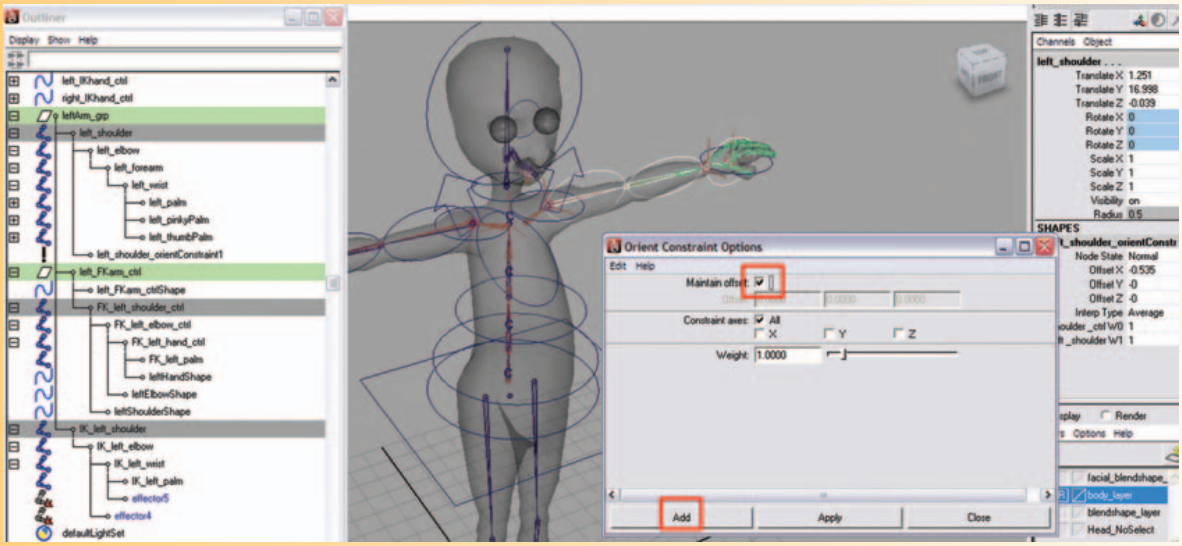
vi. In the OUTLINER, **double-click** on *group1* and rename it *rightArm_grp*.

vii. In the OUTLINER, hold down the shift key and **click** on the plus sign **(+)** next to the *leftArm_grp* to open the hierarchy and display the children. (You might want to collapse the *palm*, *pinkyPalm*, and *thumbPalm* joints so that the arm hierarchy is not as long – you can do this by clicking on the negative sign **(−)** next to them.)

viii. In the OUTLINER, **click** first on the *FK_left _shoulder_ctrl*, and while holding down the **(ctrl)** key, **click** second on the *IK_left_shoulder* and **click** third on the *left_shoulder*, then go to **[Constrain > Orient – option box]** and set the following:

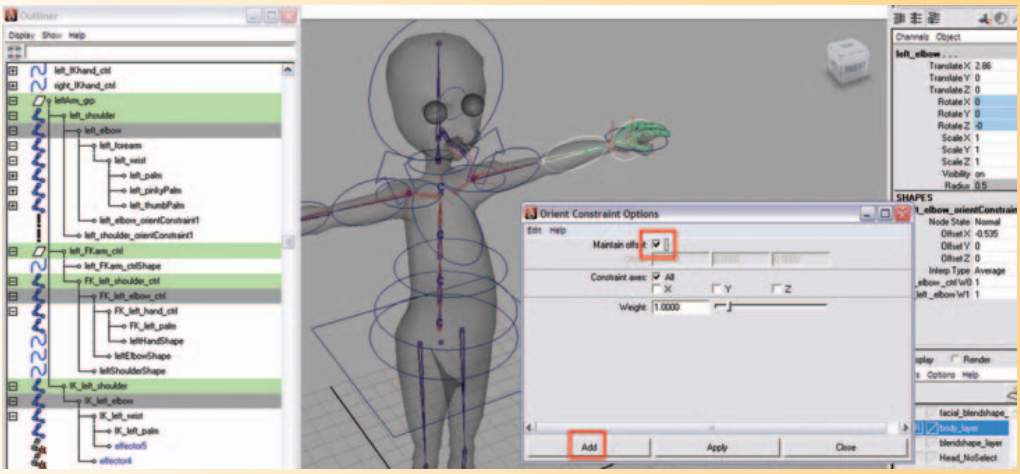
1. Go to **[Edit > Reset Settings]**.
2. Place a check mark in the box next to *Maintain Offset*.

   Then **click** "**add**".



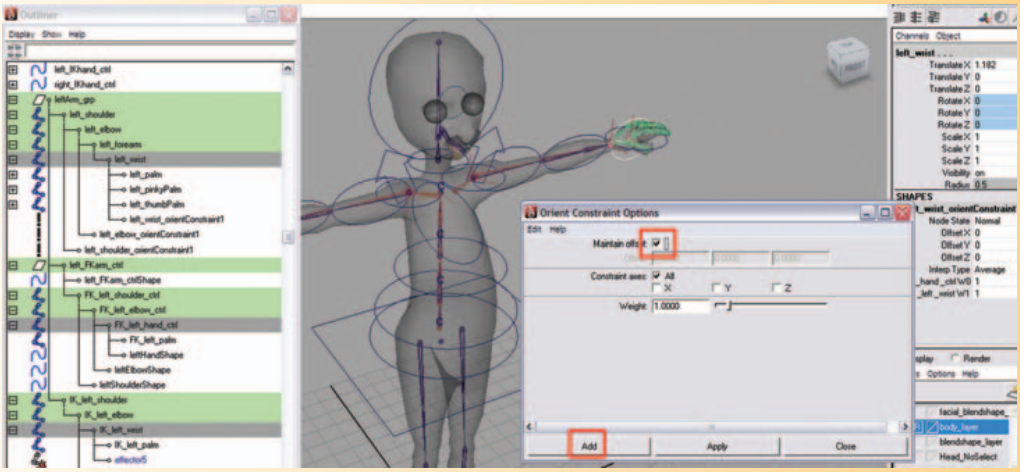
Creating an orient constraint between the *FK_left_shoulder_ctrl*, the *IK_left shoulder* (both leaders) and the *left_shoulder* (follower).

ix. In the OUTLINER, **click** first on the *FK_left_elbow_ctrl*, and while holding down the **(ctrl)** key, **click** second on the *IK_left_elbow* and **click** third on the *left_elbow*, then press the **(g)** key to repeat the last command of applying an orient constraint.



Creating an orient constraint between the *FK_left_elbow_ctrl*, the *IK_left elbow* (both leaders) and the *left_elbow* (follower).

x. In the OUTLINER, **click** first on the *FK_left_hand_ctrl*, and while holding down the **(ctrl)** key, **click** second on the *IK_left_wrist* and **click** third on the *left_wrist*, then press the **(g)** key to repeat the last command of applying an orient constraint.



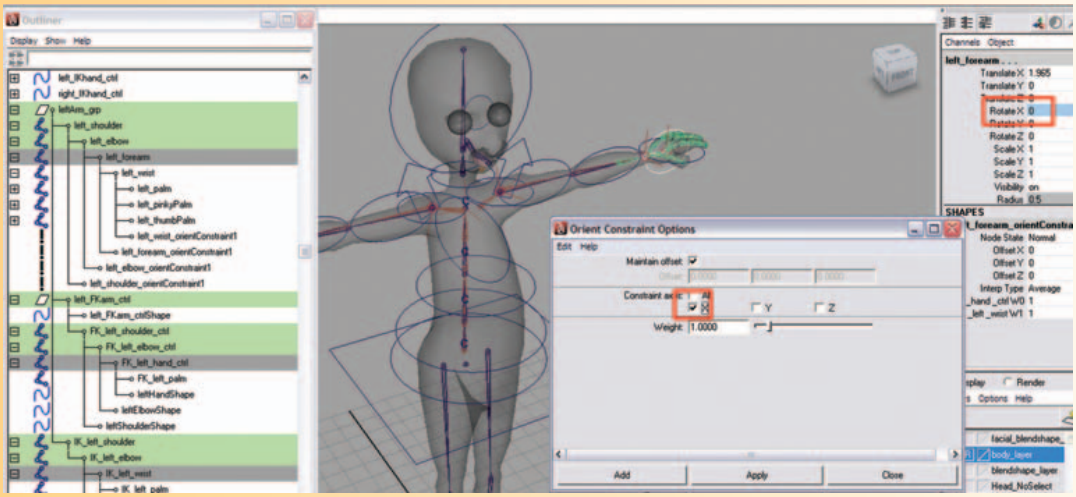
Creating an orient constraint between the *FK_left_hand_ctrl*, the *IK_left wrist* (both leaders) and the *left_wrist* (follower).

xi. In the OUTLINER, **click** first on the *FK_left_hand_ctrl*, and while holding down the **(ctrl)** key, **click** second on the *IK_left_wrist* and **click** third on the *left_forearm*, then go to **[Constrain > Orient – option box]** and set the following:

1. Go to **[Edit > Reset Settings]**.
2. Place a check mark in the box next to *Maintain Offset*.
3. Under Constraint axes: place a check mark in the box next to *X*.

   Then **click** "**add**".

(This constrain for the forearm is different from the other joints because we only want the forearm to twist with the wrist. It would look like the character's arm is broken if the forearm bent with the wrist. Because of this we only constrain the X axis.)

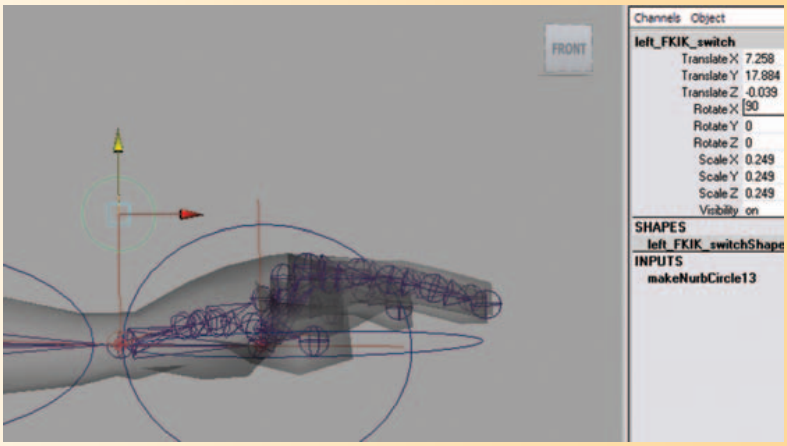


Creating an orient constraint between the *FK_left_hand_ctrl*, the *IK_left wrist* (both leaders) and the *left_forearm* (follower).

33. Repeat the constraints on the right arm.

> ! When creating the orient constraints on the right arm, MAKE SURE TO RESET THE ORIENT CONSTRAINT TO CONSTRAIN ALL AXES. To do this, open the option box and **click** the "reset tool" button as noted in the directions.

34. Create a switch to change the leader between the FK control arm and the IK control arm. To set this up, do the following:
    a. Go to **[Create > NURBS Primitives > Circle]**.
    b. In the channel box, rename *nurbsCircle1* to *left_FKIK_switch*.
    c. In PERSPECTIVE view, with the *left_FKIK_switch* selected, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *left_wrist* joint, and **click** the **MMB** and drag it slightly to snap the *left_FKIK_switch* into place.
        i. With the move tool, **click** on the Y axis **(green arrow)** and move the controller slightly above the wrist.
        ii. With the *left_FKIK_switch* selected, **select** the scale tool by pressing **(r)** and scale the *left_FKIK_switch* smaller.
        iii. With the *left_FKIK_switch* selected set the following in the channel box: RotateX: type "90".
        iv. The rotation order does not need to be changed on this controller, because rotations are not necessary for control.



Creating and positioning the left_FKIK_switch to control the ability to turn the orient constraints on and off between the control arms.

v. Duplicate the *left_FKIK_switch* by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

vi. In the OUTLINER, **double-click** on *left_FKIK_switch1* and rename it *right_FKIK_switch*.

vii. In PERSPECTIVE view, **select** the move tool by pressing **(w)** and **click** on the X axis **(red arrow)**, hold down the **(v)** key, position your cursor over the *IK_right_wrist* joint, and **click** the LMB and drag it slightly to snap the *left_FKIK_switch1* into place. (By **select**ing the X axis first, the move is constrained to that axis only.)

d. In the OUTLINER, **hold down the shift key** and **click** on the plus sign **(+)** next to the *FK_left_shoulder_ctrl* to open the hierarchy and display the children.

e. In the OUTLINER, hold down the **MMB**, **click** on the *left_FKIK_switch* and drag it onto the *left_wrist* joint. (This makes the *left_FKIK_switch* child to the *left_wrist* joint.)

f. In the OUTLINER, **hold down the shift key** and **click** on the plus sign **(+)** next to the *FK_right_shoulder_ctrl* to open the hierarchy and display the children.

g. In the OUTLINER, hold down the **MMB**, **click** on the *right_FKIK_switch* and drag it onto the *right_wrist* joint. (This makes the *right_FKIK_switch* child to the *right_wrist* joint.)

h. With the *right_FKIK_switch* selected, hold down the **(shift)** key and **select** *left_FKIK_switch*, then go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

i. With the *left_FKIK_switch* and *right_FKIK_switch* selected, go to **[Modify > Add Attribute]**. Using the default settings, **enter** the following:

i. Attribute name: type "FKIK".

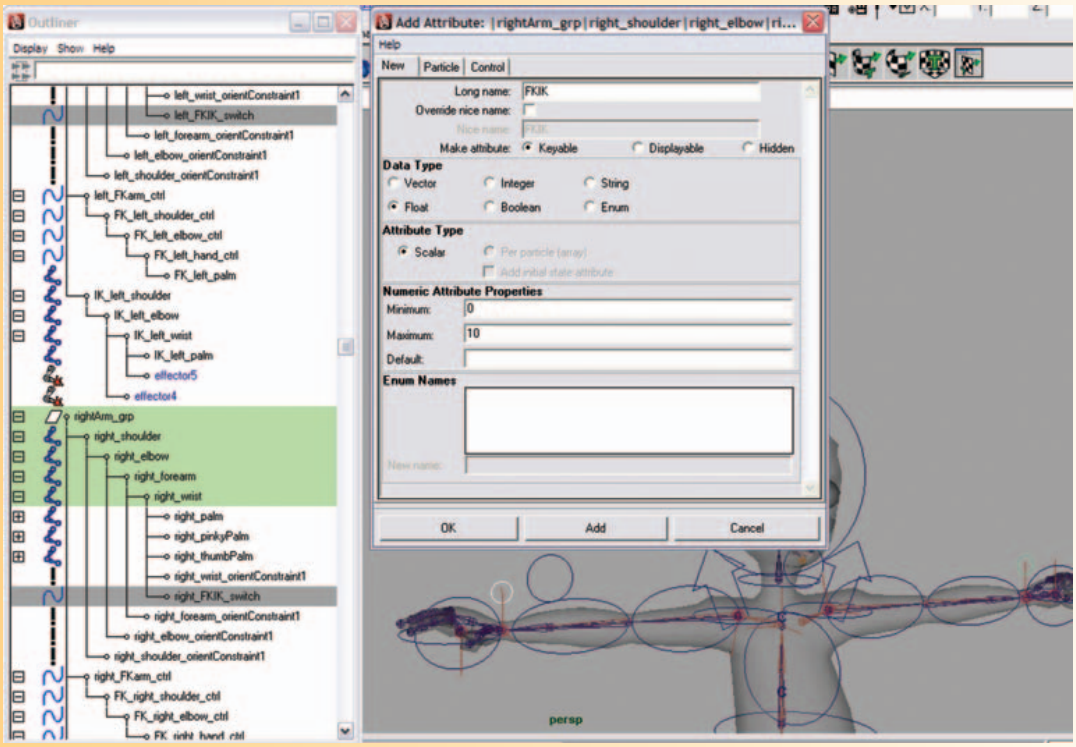
ii. Under *Numeric Attribute Properties*.

1. Minimum: type "**0**".

2. Maximum: type "**10**".

iii. **Click** "**OK**".

Data Type to Float. (Floating point numbers (or floats) have a fractional part (decimal points), allowing us to set any value between a minimum value and a maximum value which allows a smooth transition from one number to the next. The only type of attributes that we will be using in this setup is a Float.)



Adding a custom attribute to the FKIK_switch controllers using [Modify > Add Attribute].

35. Make the switch function. To set this up, we will use *Set Driven Key* to turn the constraints on and off. To sum things up, when the FKIK switch is set to 0, the FK constraints will be turned on and the IK constraints will be turned off so that the arm will follow the FK control arm. When the FKIK switch is set to 10, the IK constraints will be turned on and the FK constraints will be turned off so that the arm will follow the IK control arm. Do the following:

    a. In the OUTLINER, **select** the *left_shoulder_orientConstraint1* and go to **[Animate > Set Driven Key > Set...]**. (This places *left_shoulder_orientConstraint1* as the driven in the *Set Driven Key* window.)

        i. **Select** the *left_FKIK_switch* and **click** "Load Driver" in the *Set Driven Key* window.

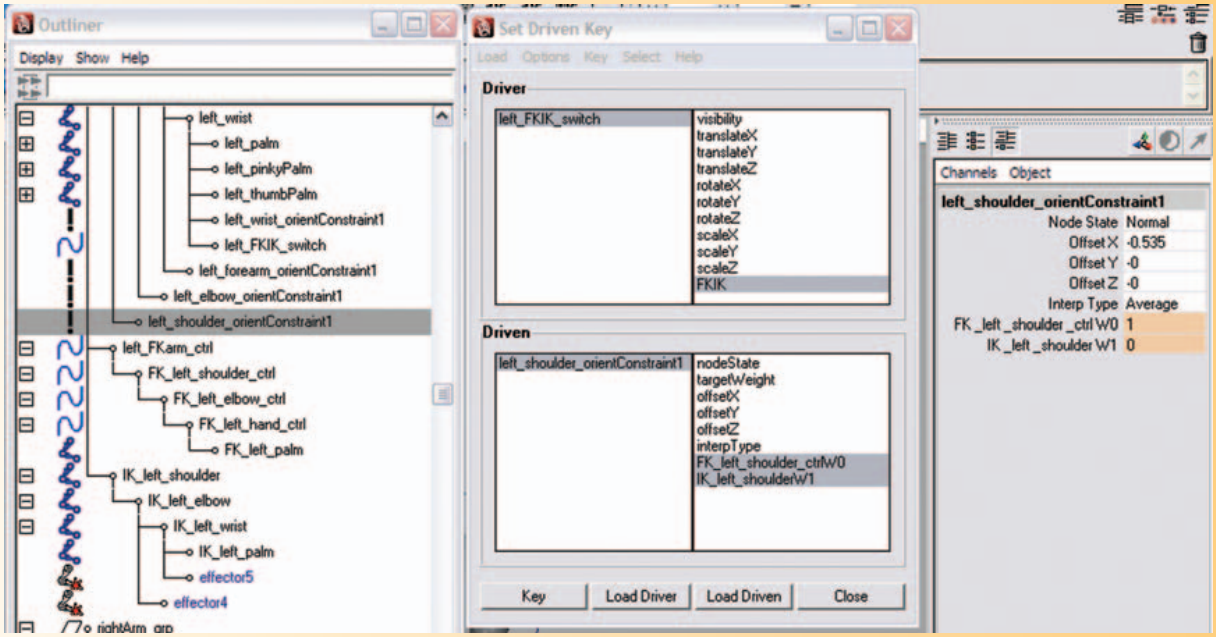
        ii. In the *Driver* section of the *Set Driven Key* window, choose "FKIK" in the right column.

iii. In the *Driven* section of the *Set Driven Key* window, choose "FK_left_shoulder_ctrlW0" in the right column, hold down the **(shift)** key and also **click** on "IK_left_shoulderW1".

iv. In the *Driven* section of the *Set Driven Key* window, **click** on *left_shoulder_orientConstraint1* to **select** it.

v. In the channel box, change *IK_left_shoulderW1* to "**0**".

vi. In the *Set Driven Key* window, **click** "**Key**". (This **click** changes the *Driven* attributes to orange in the channel box, indicating a key has been set on the shoulder constraint.)



Loading the Set Driven Key window and setting the first key so that when the FKIK_switch is set to "0", the shoulder will follow the FK_controlled shoulder.

vii. In the *Driver* section of the *Set Driven Key* window, **click** on *left_FKIK_switch* to **select** it.

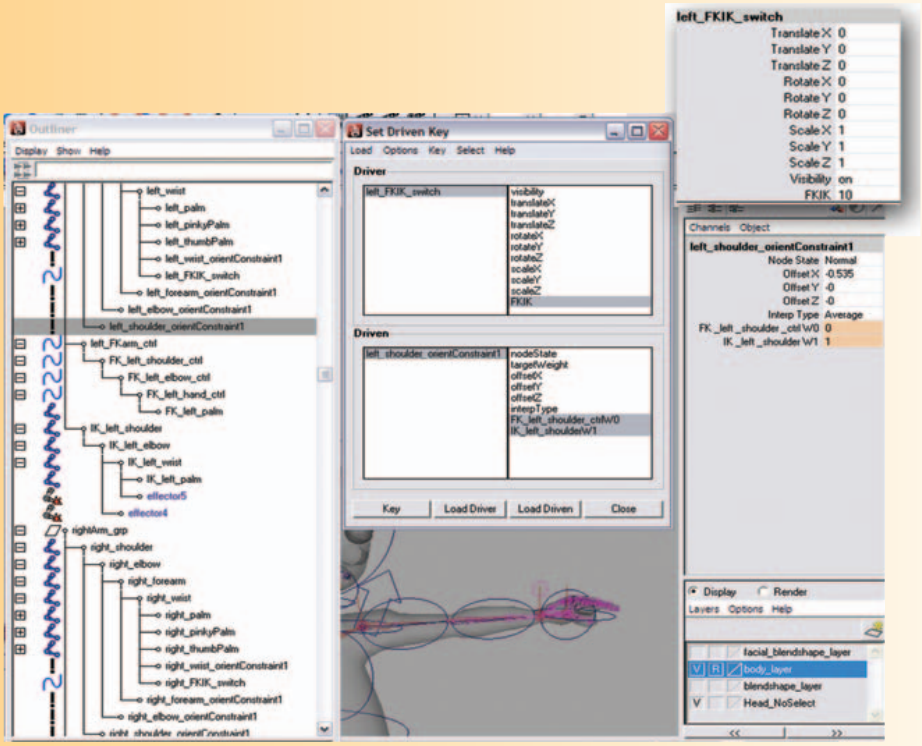
viii. In the channel box, change *FKIK* to "**10**".

ix. In the *Driven* section of the *Set Driven Key* window, **click** on *left_shoulder_orientConstraint1* to **select** it.

x. In the channel box, change *FK_left_spine_shoulder_ctrlW0* to "**0**" and *IK_left_shoulderW1* to "**1**".

xi. In the *Set Driven Key* window, **click** "**Key**".

b. Repeat these steps for the elbow, forearm, and wrist.



Setting the second key so that when the *FKIK_switch* is set to "10", the shoulder will follow the *IK_controlled shoulder*.

36. Make the switch also to control the visibility of the controllers. Since the IK controllers will only work when the FKIK switch is set to 10, we can hide them when IK has been turned off. Since the FK controllers will only work when the FKIK switch is set to 0, we can hide them when IK has been turned on. Do the following:

    a. We will start with the FK controls:

        i. In the OUTLINER, **select** the *left_FKarm_ctrl*, and then **click** "Load Driven" in the *Set Driven Key* window. (We only need the *left_FKarm_ctrl* since it is the parent of the other FK controls)

        ii. The *left_FKIK_switch* remains the *Driver* with "FKIK" chosen in the right column.

        iii. In the *Driven* section of the *Set Driven Key* window, choose "visibility" in the right column.

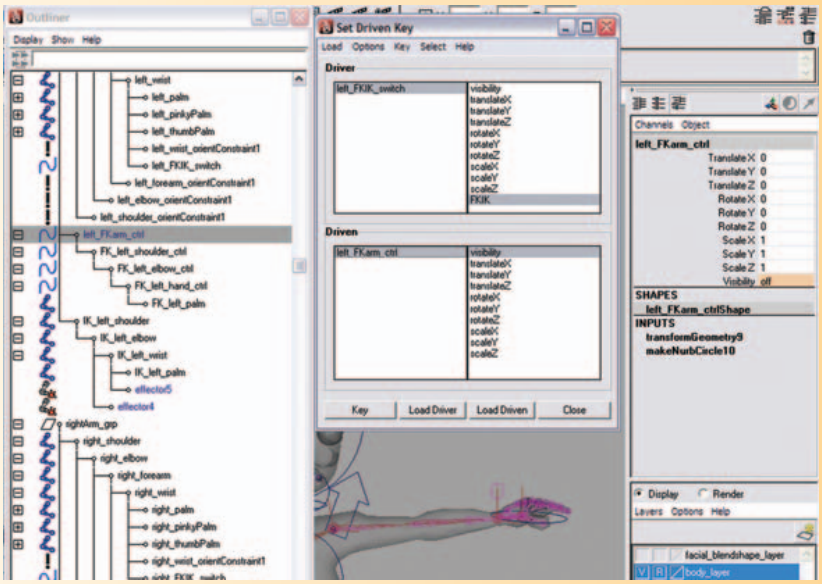
        iv. In the *Driven* section of the *Set Driven Key* window, **click** on *left_FKarm_ctrl* to **select** it.

        v. In the Channel Box, change *visibility* to "**0**" which turns the visibility off (since the last step ended with the FKIK switch on 10, we will key the FK visibility off first to save some steps.)

        vi. In the *Set Driven Key* window, **click "Key"**

        i. In the *Driver* section of the *Set Driven Key* window, **click** on *left_FKIK_switch* to **select** it.

        ii. In the channel box, change *FKIK* to "**0**".

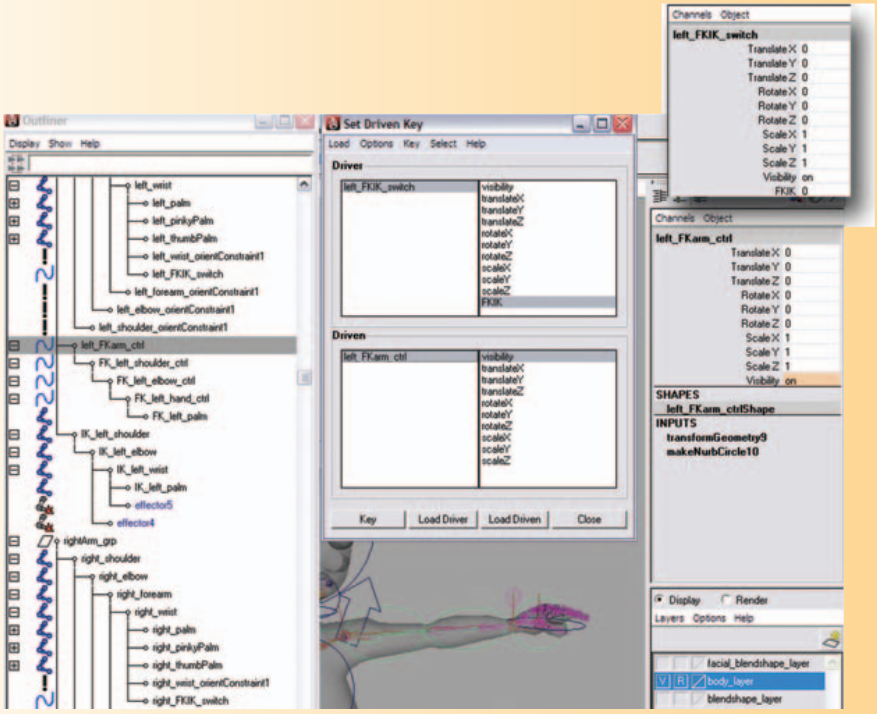


Loading the Set Driven Key window and setting the first key so that when the *FKIK_switch* is set to "10", the FK controllers are not visible.

iii. In the *Driven* section of the *Set Driven Key* window, **click** on *left_FKarm_ctrl* to **select** it.

iv. In the channel box, change *visibility* to "**1**" which turns the visibility on.

v. In the *Set Driven Key* window, **click** "**Key**".



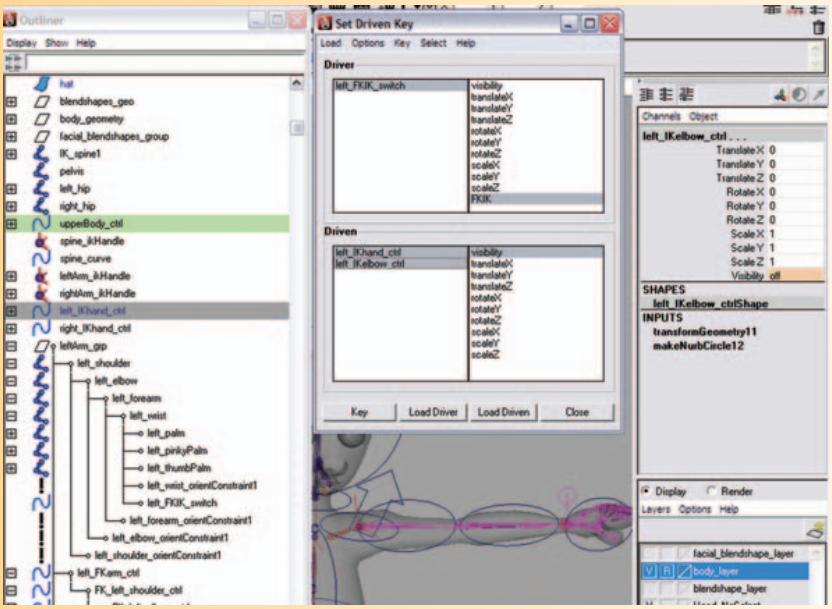
Setting the second key so that when the *FKIK_switch* is set to "0", the FK controllers are visible.

c. Repeat this process for the IK controls:

i. In the PERSPECTIVE window, **select** the *left_IKhand_ctrl*, hold down the **(ctrl)** key and **click** *left_IKelbow_ctrl*, then **click** "Load Driven" in the *Set Driven Key* window.

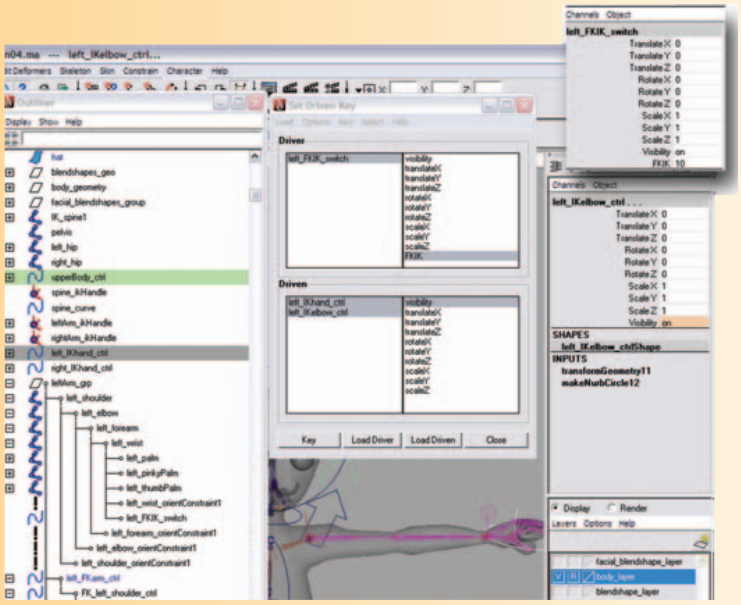
ii. The *left_FKIK_switch* remains the *Driver* with "FKIK" chosen in the right column.

iii. In the *Driven* section of the *Set Driven Key* window, choose "visibility" in the right column.



Loading the Set Driven Key window and setting the first key so that when the FKIK_switch is set to "0", the IK controllers are not visible.

iv. In the *Driven* section of the *Set Driven Key* window, **click** on *left_IKarm_ctrl*, hold down the **(shift)** key and also **click** on *left_IKelbow_ctrl* to **select** them.



Setting the second key so that when the FKIK_switch is set to "10", the IK controllers are visible.

v. In the channel box, change *visibility* to "**0**".

vi. In the *Set Driven Key* window, **click** "**Key**".

vii. In the *Driver* section of the *Set Driven Key* window, **click** on *left_FKIK_switch* to **select** it.

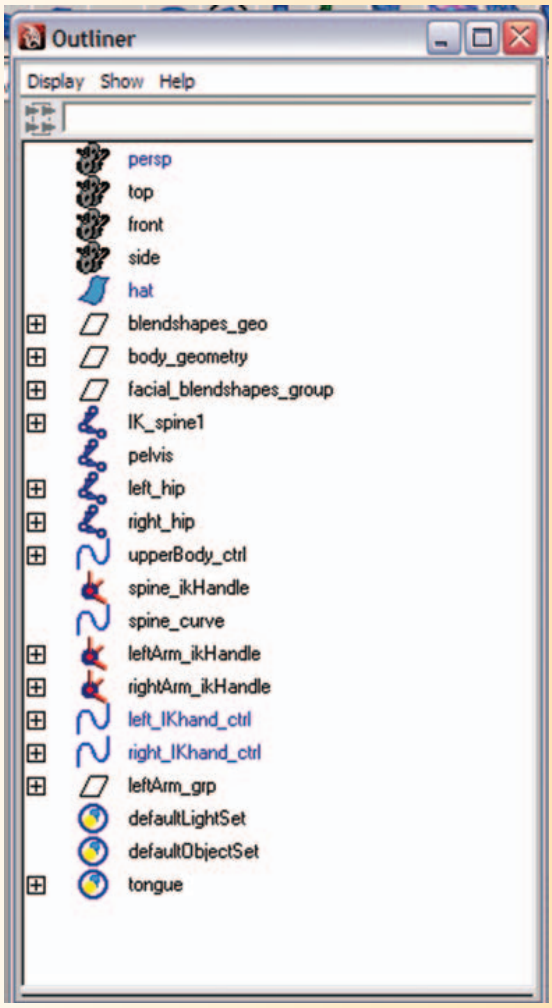
viii. In the channel box, change *FKIK* to "**10**".

ix. In the *Driven* section of the *Set Driven Key* window, **click** on *left_IKarm_ctrl*, hold down the **(shift)** key and also **click** on *left_IKelbow_ctrl* to **select** them.

x. In the channel box, change *visibility* to "**1**".

xi. In the *Set Driven Key* window, **click** "**Key**".

xii. Repeat making the switch function for the right arm.



The new OUTLINER hierarchy after integrating the arms into the existing spine controls.

37. Integrate the FK and IK arms into the existing shoulder controls by doing the following:

a. In the PERSPECTIVE window, **click** on the *left_FKIK_switch*, hold down the **(shift)** key and **click** the *right_FKIK_switch*. In the channel box, set the FKIK attribute to "**0**" to turn IK off.

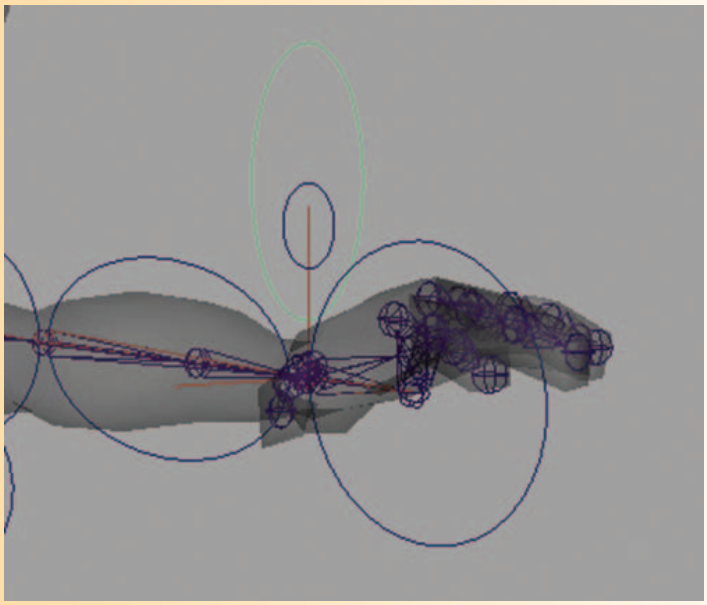
b. In the OUTLINER, **click** on the *leftArm_grp*, hold down the **(shift)** key on the keyboard and in the PERSPECTIVE window, **click** on the *left_clavicle_ctrl* and press **(p)** on the keyboard to parent them.

c. In the OUTLINER, **click** on the *rightArm_grp*, hold down the **(shift)** key on the keyboard and in the PERSPECTIVE window, **click** on the *right_clavicle_ctrl* and press **(p)** on the keyboard to parent them.

38. Save your scene file. Name your scene *06_asgn04.ma*.

Assignment 6.5: Creating a Control System for the Fingers

1. Open Maya and set your project.
   a. From your computer's desktop, go to **[Start > Programs]** and **select** Maya.
   b. Once Maya is open go to **[File > Project > Set...]** and browse to your project folder then **click** "**OK**".
2. Open your last saved file: Go to **[File > Open]** and **select** *06_asgn04.ma*.
3. Continue working in X-ray mode.
4. Make sure that your geometry layer is set to **R** for reference so that you are unable to **select** the geometry by mistake when working.
5. To make selection easier open your OUTLINER by going to **[Windows > Outliner]**.
6. Create a control system for the fingers by doing the following:
   a. Go to **[Create > NURBS Primitives > Circle]**.
   b. In the channel box, rename *nurbsCircle1* to *left_finger_ctrl*.
   c. In PERSPECTIVE view, with the *left_finger_ctrl* selected, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *left_wrist* joint, and **click** the **MMB** and drag it slightly to snap the *left_finger_ctrl* into place.
      i. With the *left_finger_ctrl* selected, set the following in the channel box: RotateZ: type "**90**".
      ii. With the move tool, **click** on the Y axis **(green arrow)** and move the controller slightly above the wrist, around the *left_FKIK_switch*.
      iii. With the *left_finger_ctrl* selected, **select** the scale tool by pressing **(r)** and scale the *left_finger_ctrl* larger than the *left_FKIK_switch*.
      iv. The rotation order does not need to be changed on this controller, because rotations are not necessary for control.



Creating and positioning the *left_finger_ctrl*.

v. Duplicate the *left_finger_ctrl* by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

vi. In the OUTLINER, **double-click** on *left_finger_ctrl1* and rename it *right_finger_ctrl*.

vii. In PERSPECTIVE view, **select** the move tool by pressing **(w)** and **click** on the X axis **(red arrow)**, hold down the **(v)** key, position your cursor over the *right_finger_ctrl*, and **click** the **MMB** and drag it slightly to snap the *right_finger_ctrl* into place. (By **select**ing the X axis first, the move is constrained to that axis only.)

d. In the PERSPECTIVE window, **select** the *left_finger_ctrl*, hold down the **(shift)** key and **click** the *left_FKIK_switch*, and press **(p)** to parent.

e. In the PERSPECTIVE window, **select** the *right_finger_ctrl*, hold down the **(shift)** key and **click** the *right_FKIK_switch*, and press **(p)** to parent.

f. With the *right_finger_ctrl* selected, hold down the **(shift)** key and **select** *left_finger_ctrl*, then go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

7. Add attributes to the finger control for the finger movements by doing the following:

a. With the *left_finger_ctrl* and *right_finger_ctrl* selected, go to **[Modify > Add Attributes]** and using the default settings, **enter** the following:

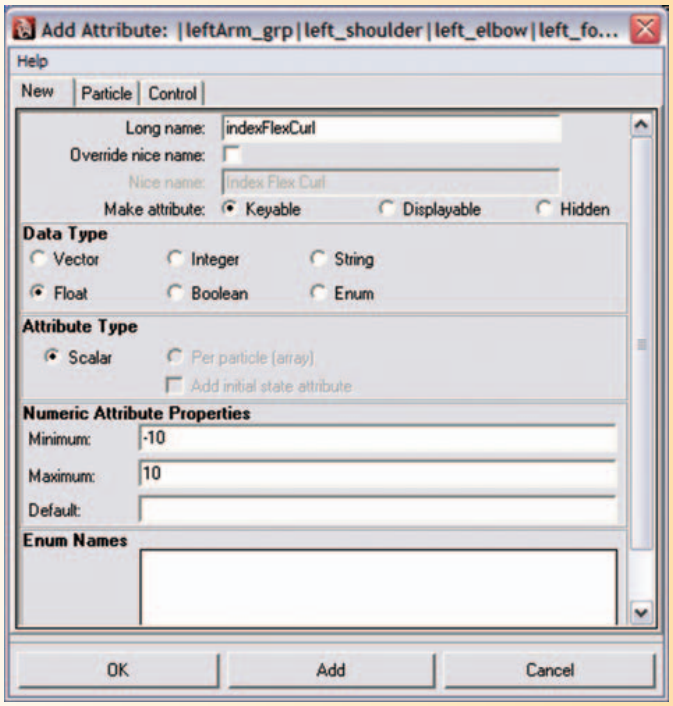
i. Attribute name: type "**indexFlexCurl**".

ii. Under *Numeric Attribute Properties*

1. Minimum: type "−10".

2. Maximum: type "**10**".

3. **Click** "**Add**".



Adding custom attributes to the fingers using [Modify > Add Attribute].

b. **Enter** the following:
   i. Attribute name: type "**middleFlexCurl**".
   ii. Under *Numeric Attribute Properties*
      1. Minimum: type "−10".
      2. Maximum: type "**10**".
      3. **Click** "**Add**"

c. **Enter** the following:
   i. Attribute name: type "**ringFlexCurl**".
   ii. Under *Numeric Attribute Properties*
      1. Minimum: type "−10".
      2. Maximum: type "**10**".
      3. **Click** "**Add**".

d. **Enter** the following:
   i. Attribute name: type "**pinkyFlexCurl**".
   ii. Under *Numeric Attribute Properties*
      1. Minimum: type "−10".
      2. Maximum: type "**10**".
      3. **Click** "**Add**".

e. **Enter** the following:
   i. Attribute name: type "**thumbFlexCurl**".
   ii. Under *Numeric Attribute Properties*
      1. Minimum: type "−10".
      2. Maximum: type "**10**".
      3. **Click** "**Add**".

f. **Enter** the following:
   i. Attribute name: type "**thumbSpread**".
   ii. Under *Numeric Attribute Properties*
      1. Minimum: type "−10".
      2. Maximum: type "**10**".
      3. **Click** "**Add**".

g. **Enter** the following:
   i. Attribute name: type "**fingerSpread**".
   ii. Under *Numeric Attribute Properties*
      1. Minimum: type "−10".
      2. Maximum: type "**10**".
      3. **Click** "**OK**".

h. **Enter** the following:

i. Attribute name: type "**grab**".

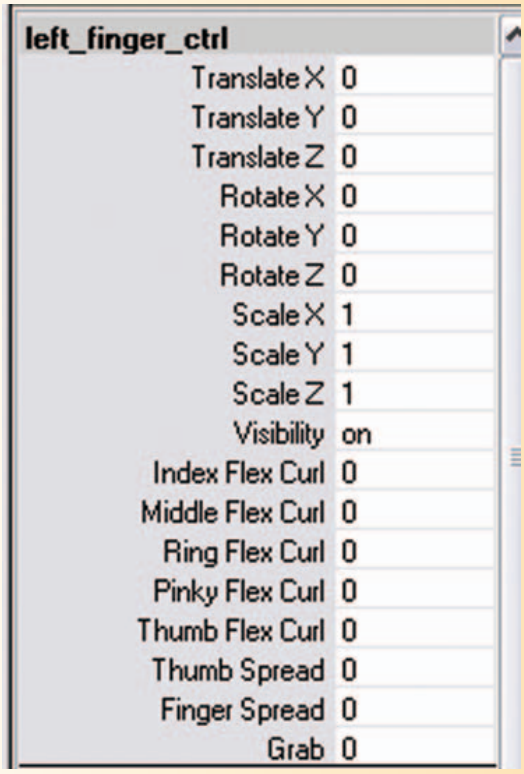
ii. Under *Numeric Attribute Properties*

1. Minimum: type "**0**".

2. Maximum: type "**10**".

3. **Click** "**Add**".

These are the attributes that you will control with Set Driven Key. They now show up in the channel box for the *left_finger_ctrl*. For more control over the fingers, you might even want break apart the finger motion into two separate attributes for each finger: *a bend* attribute for the first knuckle of the finger and *a curl* attribute for the last two knuckles. If you make a mistake, you can go to **[Modify > Edit Attribute]** to make changes.



The left_finger_ctrl in the channel box, with all of the finger attributes added.

! When creating the poses for the fingers using Set Driven Key, you can do all the **select**ing and manipulating in the work area (PERSPECTIVE window) or use the OUTLINER or Hypergraph if **select**ing the joints in the view panel becomes too tedious. The joints can be selected from inside the Set Driven Key editor as well but make sure to reselect them all in the "Driven" section before pressing the "**Key**" button in the set driven window.

8. Use Set Driven Key to add functionality to the attributes by doing the following:

a. In the OUTLINER, **select** the *left_index1* joint, hold down the **(ctrl)** key and **click** *left_index2* joint and *left_index3* joint and go to **[Animate > Set Driven Key > Set...]**. (This places the joints as the driven in the *Set Driven Key* window.)

b. Add functionality to the *indexFlexCurl* attribute.

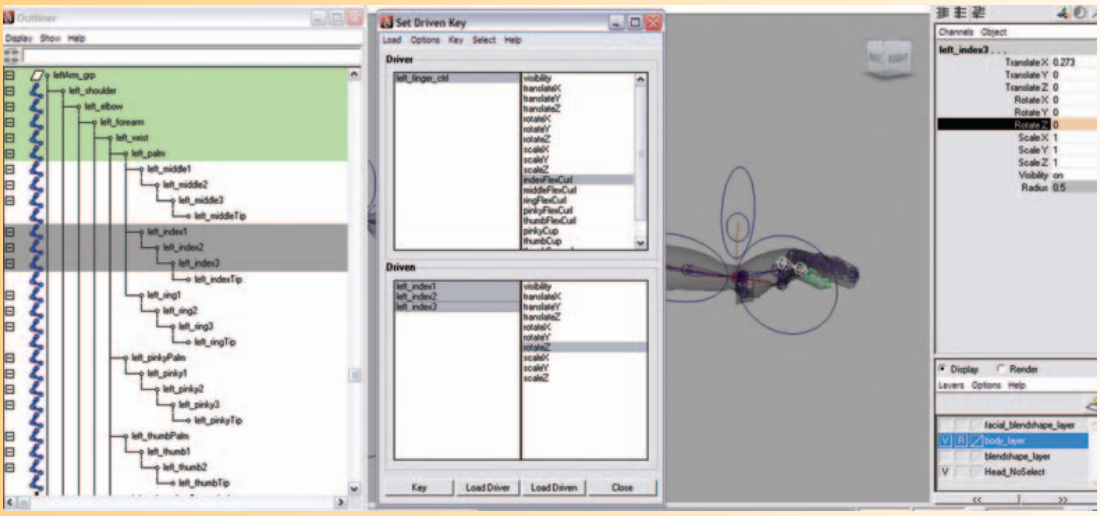
i. **Select** the *left_finger_ctrl* and **click** "Load Driver" in the *Set Driven Key* window.

ii. In the *Driver* section of the *Set Driven Key* window, choose "**indexFlexCurl**" in the right column.

iii. In the *Driven* section of the *Set Driven Key* window, **click** on *left_index1* joint, hold down the **(ctrl)** key and **click** *left_index2* joint and *left_index3* to **select** them.

iv. In the *Driven* section of the *Set Driven Key* window, choose "rotate Z" in the right column.

v. In the *Set Driven Key* window, **click** "**Key**". (This sets a default finger pose position at attribute value of 0.)



Loading the Set Driven Key window and setting the first key so that when the indexFlexCurl attribute is set to "0", the index finger is in the default (original) position.

vi. In the *Driver* section of the *Set Driven Key* window, **click** on *left_finger_ctrl* to **select** it.

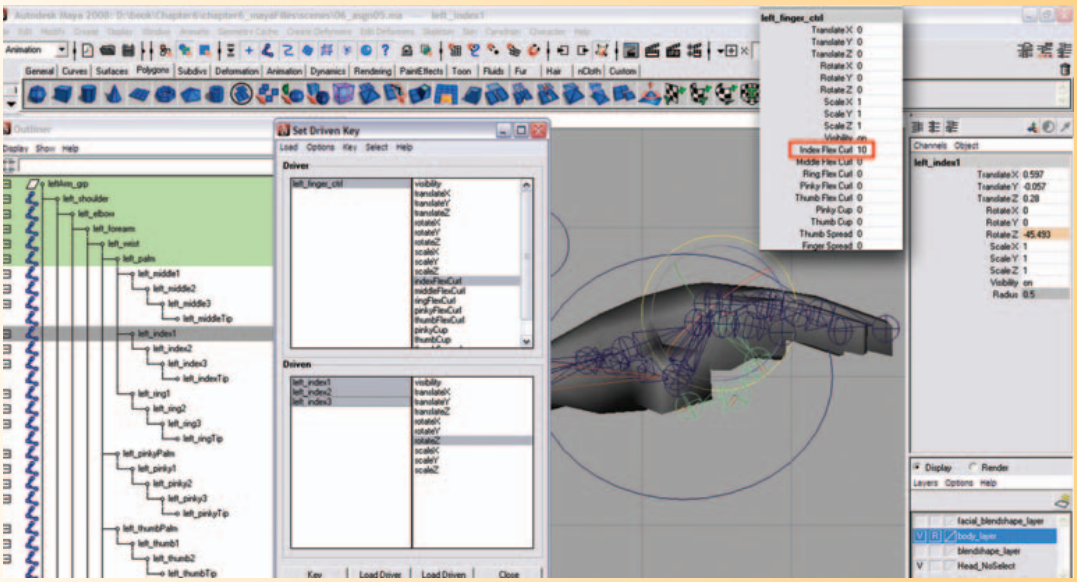
vii. In the channel box, change *indexFlexCurl* to "**10**".

viii. In the PERSPECTIVE window, **select** the rotate tool by pressing the **(e)** key, rotate the index finger joints (index1, index2, and index3) along the Z axis (the blue ring) into a curled bent position. (Notice how your finger bends when you make a fist and try to mimic the position.)

ix. In the *Set Driven Key* window, **click** "**Key**". (This sets a keyed pose of the finger in a bent position at attribute value of "**10**".)

x. In the *Driver* section of the *Set Driven Key* window, **click** on *left_finger_ctrl* to **select** it.

xi. In the channel box, change *indexFlexCurl* to "−10".

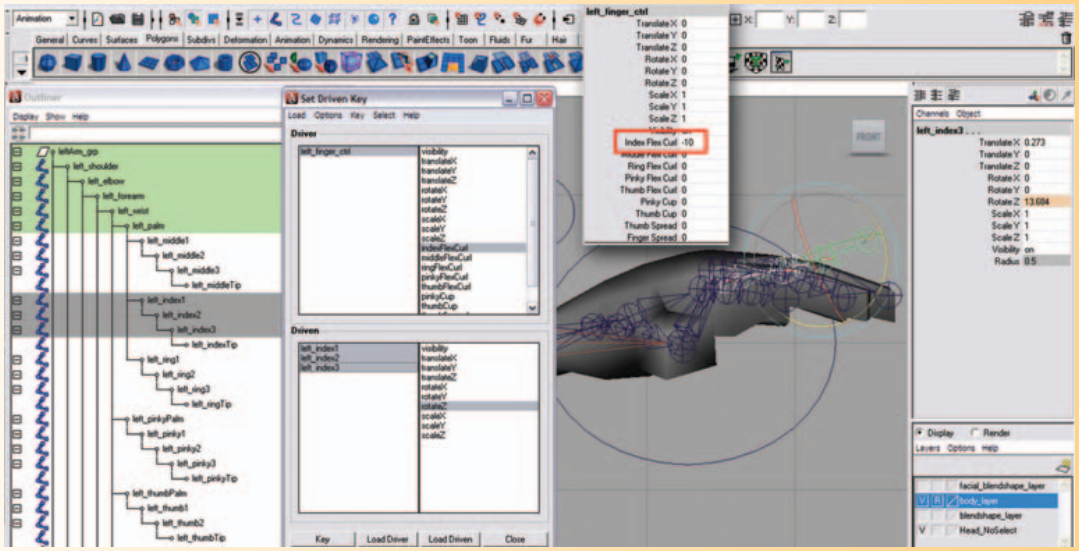


Setting the second key so that when the indexFlexCurl is set to "10", the index finger is in the curled position.

xii. In the PERSPECTIVE window, **select** the rotate tool by pressing the **(e)** key, rotate the index finger joints (index1, index2, and index3) along the Z axis (the blue ring) into a flexed position. (Notice how your finger flexes when you stretch your hand and try to mimic the position.)

xiii. In the *Set Driven Key* window, **click** "**Key**". (This **click** changes the *Driven* attributes to orange in the channel box, indicating a key has been set.)

xiv. To test and see if the control works, **click** on the word *indexFlexCurl* in the channel box, then in the PERSPECTIVE window, **MMB click** and drag your mouse left to right. You should see your character's index finger flex and curl.



Setting the third key so that when the indexFlexCurl is set to "−10", the index finger is in the flexed position.

xv. In the channel box, change *indexFlexCurl* to "**0**".

> **!** This exact procedure is repeated for the middle finger, the ring finger, the pinky, and the thumb. The important thing to remember is to make sure you change the driver attribute and reload the new joints as your driven.

c. Repeat these steps for the *middleFlexCurl, ringFlexCurl, pinkyFlexCurl,* and *thumbFlexCurl* attributes.

> **!** This procedure is similar for the thumbSpread and fingerSpread attributes. The main differences are the direction of the rotations and that multiple joints need to be rotated to achieve the position. Each position will be slightly different depending on how the hand was modeled.

d. Repeat these steps for the *thumbSpread* attribute.

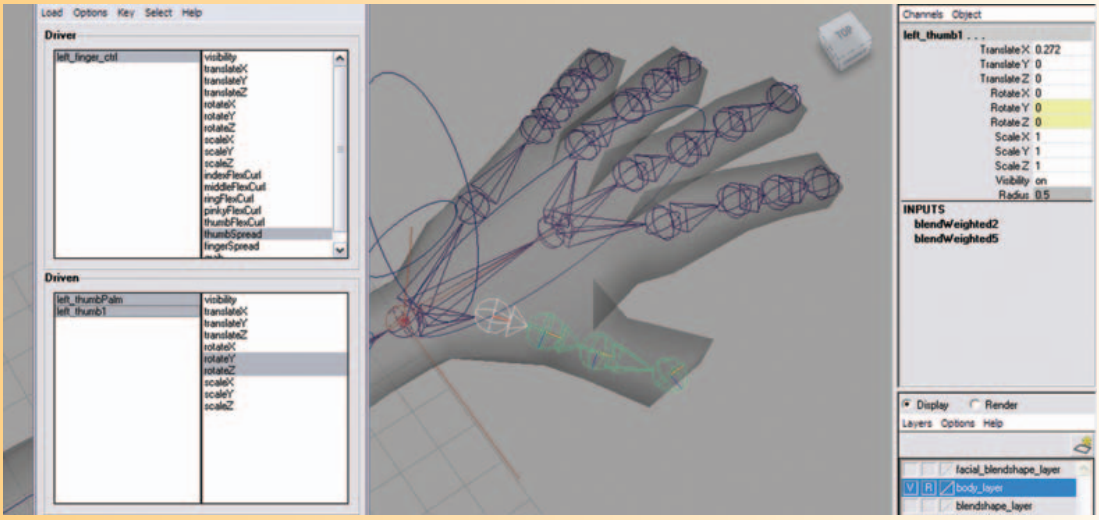
i. In the *Driver* section of the *Set Driven Key* window, choose "thumbSpread" in the right column.

ii. **Select** the **select** the *left_thumbPalm* joint, hold down the **(ctrl)** key and **click** *left_thumb1* joint, then **click** "Load Driven" in the *Set Driven Key* window.

iii. In the *Driven* section of the *Set Driven Key* window, **click** on *left_thumbPalm* joint, hold down the **(ctrl)** key and **click** *left_thumb1* joint to **select** them.

iv. In the *Driven* section of the *Set Driven Key* window, **click** "rotate Y, then hold down the **(ctrl)** key and **click** rotate Z" in the right column.

v. In the *Set Driven Key* window, **click** "**Key**". (This sets a default finger pose position at attribute value of 0.)



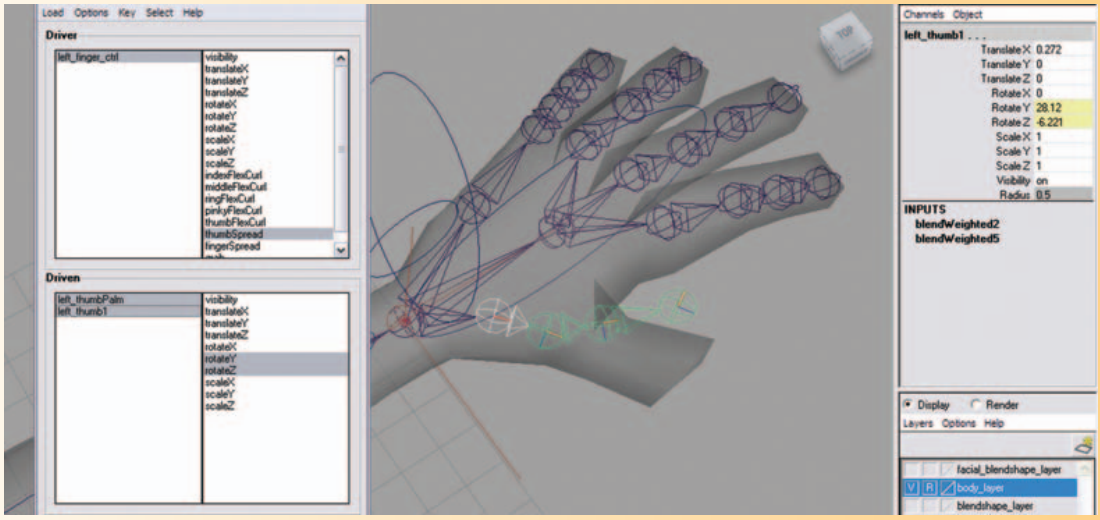
Loading the Set Driven Key window and setting the first key so that when the thumbSpread attribute is set to "0", the thumb is in the default (original) position.

vi. In the *Driver* section of the *Set Driven Key* window, **click** on *left_finger_ctrl* to **select** it.

vii. In the channel box, change *thumbSpread* to "**10**".

viii. In the PERSPECTIVE window, **select** the rotate tool by pressing the **(e)** key, rotate the thumb finger joints (*thumbPalm* and *thumb1*) along the Y or Z axis into a position next to the index finger.

ix. In the *Set Driven Key* window, **click** "**Key**". (This sets a keyed pose of the thumb in a cupped position at attribute value of "**10**".)



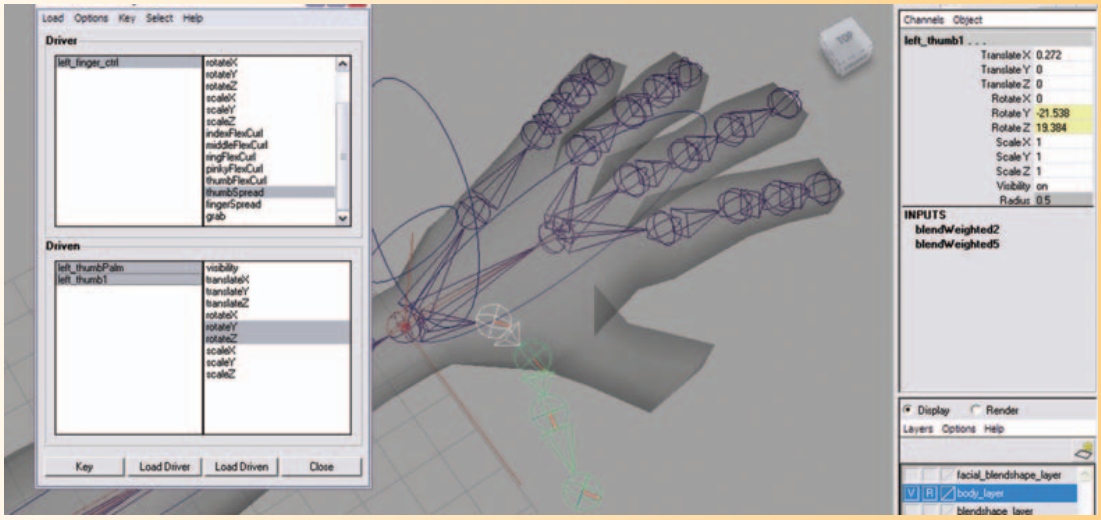
Setting the second key so that when the thumbSpread is set to "10", the thumb is near the index finger in a closed position.

x. In the *Driver* section of the *Set Driven Key* window, **click** on *left_finger_ctrl* to **select** it.

xi. In the channel box, change *thumbSpread* to "−10".

xii. In the PERSPECTIVE window, **select** the rotate tool by pressing the **(e)** key, rotate the thumb finger joints (*thumbPalm* and *thumb1*) along the Y or Z axis into a position away from the index finger.

xiii. In the *Set Driven Key* window, **click** "**Key**". (This sets a keyed pose of the thumb in a cupped position at attribute value of "−10".)



Setting the third key so that when the thumbSpread is set to "−10", the thumb is away from the index finger in a spread position.

xiv. To test and see if the control works, **click** on the word *thumbSpread* in the channel box, then in the PERSPECTIVE window, **MMB click** and drag your mouse left to right.

xv. In the channel box, change *thumbSpread* to "**0**".

e. Repeat these steps for the *fingerSpread* attribute.

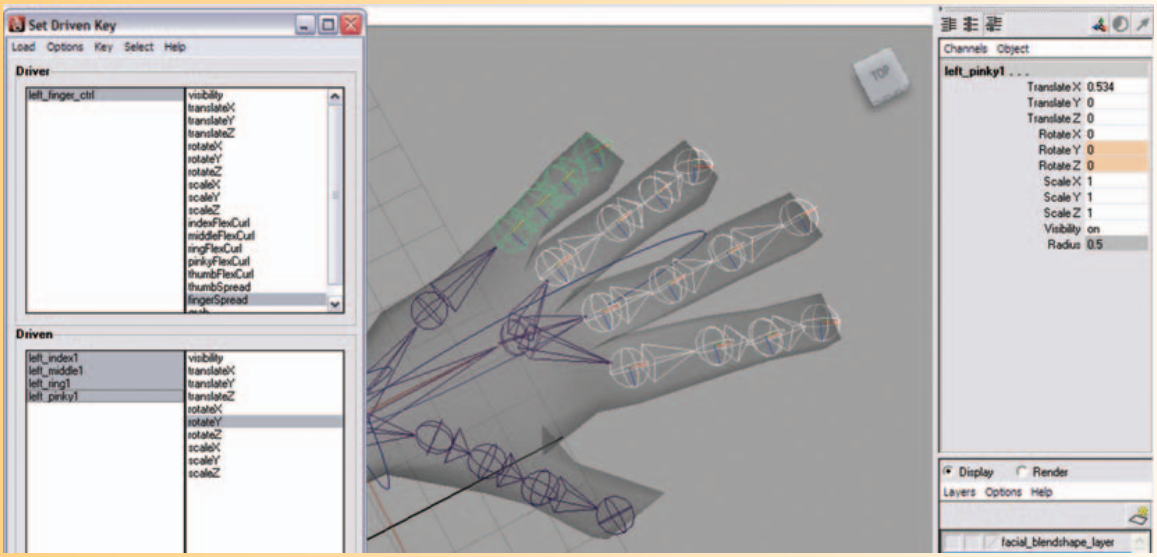
i. In the *Driver* section of the *Set Driven Key* window, choose "fingerSpread" in the right column.

ii. **Select** the **select** the *left_index1* joint, hold down the **(ctrl)** key and **click** the *left_middle1* joint, the *left_ring1* joint, and the *left_pinky1* joint, **click** "Load Driven" in the *Set Driven Key* window.

iii. In the *Driven* section of the *Set Driven Key* window, **click** the *left_index1* joint, hold down the **(ctrl)** key and **click** the *left_middle1* joint, the *left_ring1* joint, *and* the *left_pinky1* joint to **select** them.

iv. In the *Driven* section of the *Set Driven Key* window, choose "rotate Y" in the right column.

v. In the *Set Driven Key* window, **click** "**Key**".



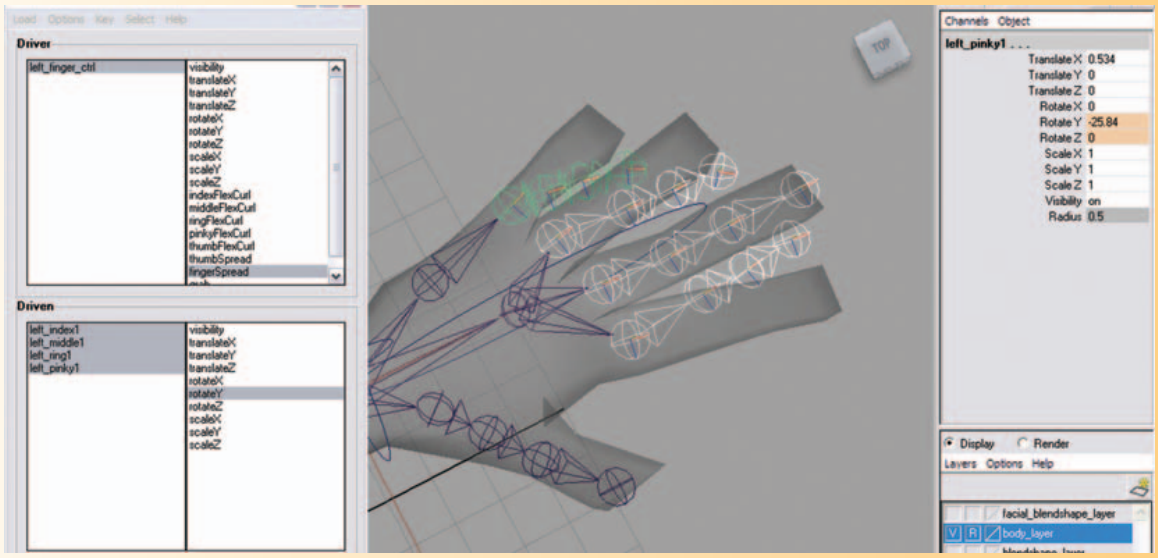
Loading the Set Driven Key window and setting the first key so that when the fingerSpread attribute is set to "0", the hand is in the default (original) position.

vi. In the *Driver* section of the *Set Driven Key* window, **click** on *left_finger_ctrl* to **select** it.

vii. In the channel box, change *fingerSpread* to "**10**".

viii. In the PERSPECTIVE window, **select** the rotate tool by pressing the **(e)** key, rotate the finger joints (*left_index1*, *left_middle1*, *left_ring1*, and *left_pinky1*) along the Y axis (the green ring) into a squeezed closed position where all of the fingers would be touching.

ix. In the *Set Driven Key* window, **click** "**Key**".



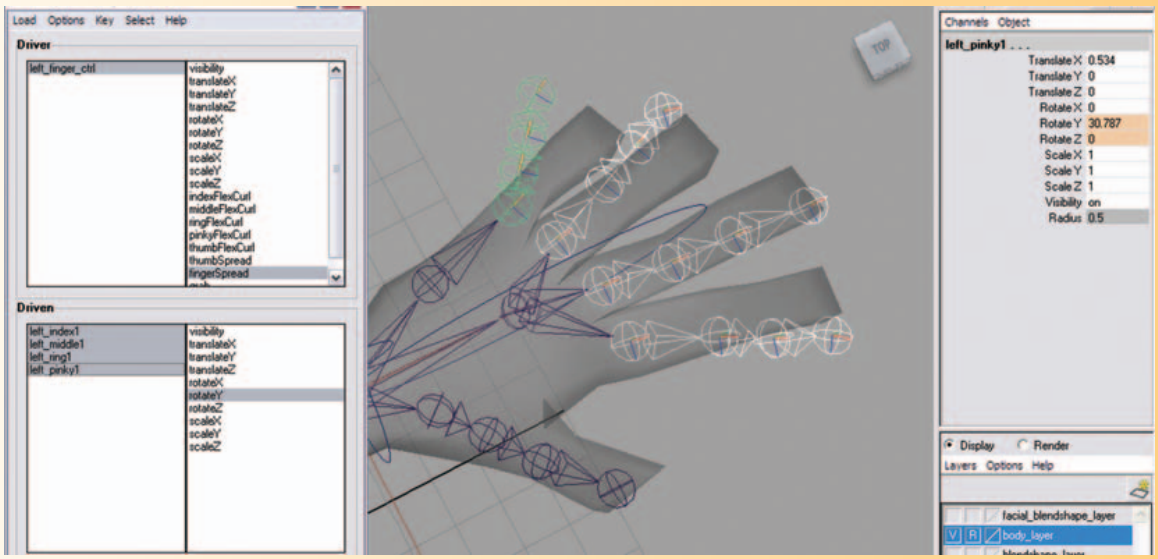
Setting the second key so that when the fingerSpread is set to "10", the fingers are in a closed position.

x. In the *Driver* section of the *Set Driven Key* window, **click** on *left_finger_ctrl* to **select** it.

xi. In the channel box, change *fingerSpread* to "−10".

xii. In the PERSPECTIVE window, **select** the rotate tool by pressing the **(e)** key, rotate the finger joints (*left_index1*, *left_middle1*, *left_ring1*, and *left_pinky1*) along the Y axis (the green ring) into a spread open position.

xiii. In the *Set Driven Key* window, **click** "**Key**".



Setting the third key so that when the fingerSpread is set to "−10", the fingers are in a spread open position.

xiv. To test and see if the control works, **click** on the word *fingerSpread* in the channel box, then in the PERSPECTIVE window, **MMB click** and drag your mouse left to right. You should see your character's fingers spread and close.

xv. In the channel box, change *fingerSpread* to "**0**".

! This last position is for grabbing. It's a position where the thumb and pinky are cupped forward with the rest of the hand. Think about a pose where the hand is reaching for a doorknob.

f. Repeat these steps for the *grab* attribute.

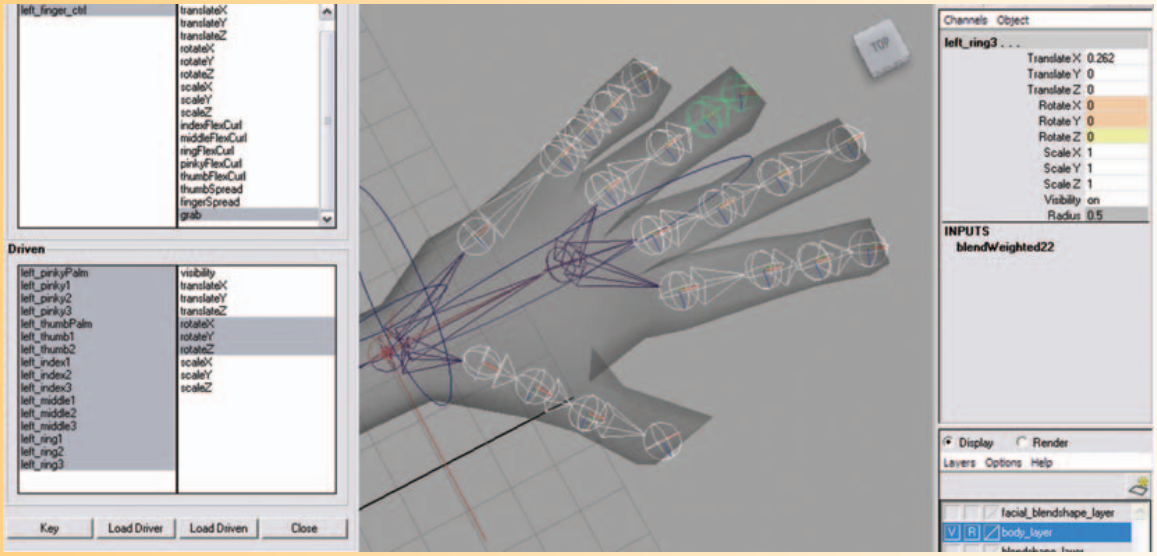
i. In the *Driver* section of the *Set Driven Key* window, choose "grab" in the right column.

ii. **Select** all of the joints in the hand (except for the Tip joints) and **click** "Load Driven" in the *Set Driven Key* window.

iii. In the *Driven* section of the *Set Driven Key* window, **click** on all of the joints to **select** them.

iv. In the *Driven* section of the *Set Driven Key* window, **click** "rotate X, then hold down the **(ctrl)** key and **click** rotate Y and rotate Z" in the right column.

v. In the *Set Driven Key* window, **click** "**Key**". (This sets a default finger pose position at attribute value of 0.)



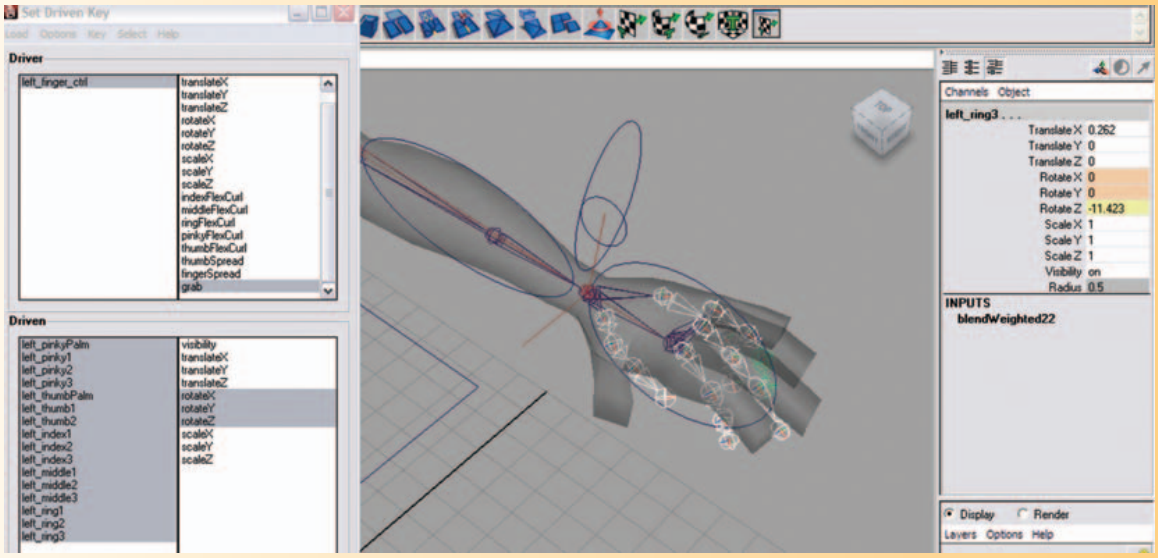
Loading the Set Driven Key window and setting the first key so that when the grab attribute is set to "0", the hand is in the default (original) position.

vi. In the *Driver* section of the *Set Driven Key* window, **click** on *left_finger_ctrl* to **select** it.

vii. In the channel box, change *grab* to "**10**".

viii. In the PERSPECTIVE window, **select** the rotate tool by pressing the **(e)** key, and rotate the finger joints along the X, Y, or Z axis into a cupped bent position. (The goal is to make it so that the hand can grab something, so they would be rotated slightly inward toward the palm.)

ix. In the *Set Driven Key* window, **click** "**Key**". (This sets a keyed pose of the fingers in a cupped position at attribute value of "**10**".)



Setting the second key so that when the grab is set to "10", the fingers are in a cupped position.

x. To test and see if the control works, **click** on the word *pinkyCup* in the channel box, then in the PERSPECTIVE window, **MMB click** and drag your mouse left to right.

xi. In the channel box, change *pinkyCup* to "**0**".

9. Repeat adding functionality to the attributes for the right fingers.

10. Save your scene file. Name your scene *06_asgn05.ma*.

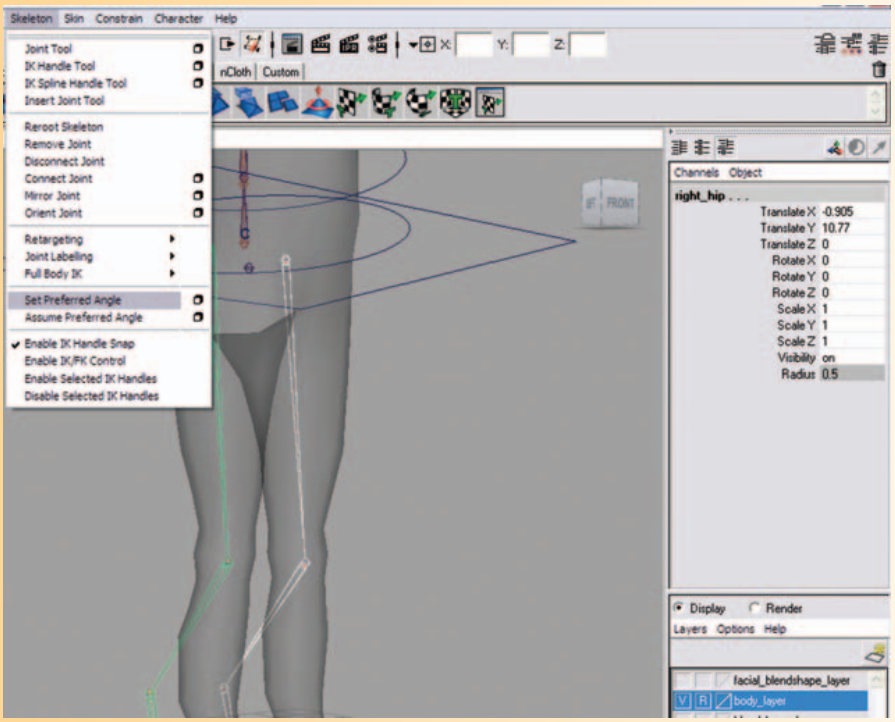
Assignment 6.6: Creating a Control System for the Legs and Feet

For the leg control, we will only be setting up an IK control system since most of the time our characters will be walking on something. If you think you need an FK control system, you can follow the setup for the arms and adapt it to the legs.

1. Open Maya and set your project.
   a. From your computer's desktop, go to **[Start > Programs]** and **select** Maya.
   b. Once Maya is open, go to **[File > Project > Set...]** and browse to your project folder then **click** "**OK**".
2. Open your last saved file: Go to **[File > Open]** and **select** *06_asgn05.ma*.
3. Continue working in X-ray mode.
4. Make sure that your geometry layer is set to **R** for reference so that you are unable to **select** the geometry by mistake when working.
5. To make selection easier open your OUTLINER by going to **[Windows > Outliner]**.

6. Create the IK in the leg by doing the following:
   a. Set a preferred angle in the left leg by doing the following:
      i. **Select** the *left_knee* joint and the *right_knee* joint in the channel box set the following: RotateY: type "**40**".
      ii. **Select** the *left_hip* joint, then go to **[Skeleton > Set Preferred Angle]**.
      iii. **Select** the *left_knee* joint and in the channel box set the following: RotateY: type "**0**" (this will make the knee straight again).

   (We must first set a preferred angle in the leg so that Maya knows which direction to bend the leg when we run the IK solver through the joints.)

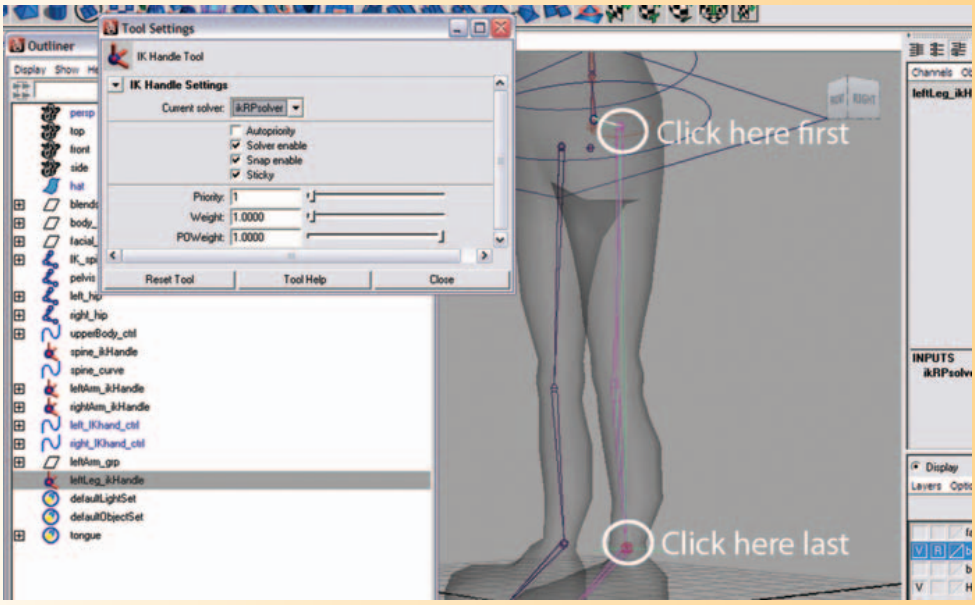


Setting a preferred angle on the legs.

   b. Go to **[Skeleton > IK Handle Tool – option box]** and set the following:
      i. **Click** "reset tool" then under *IK Handle Settings* change the following:

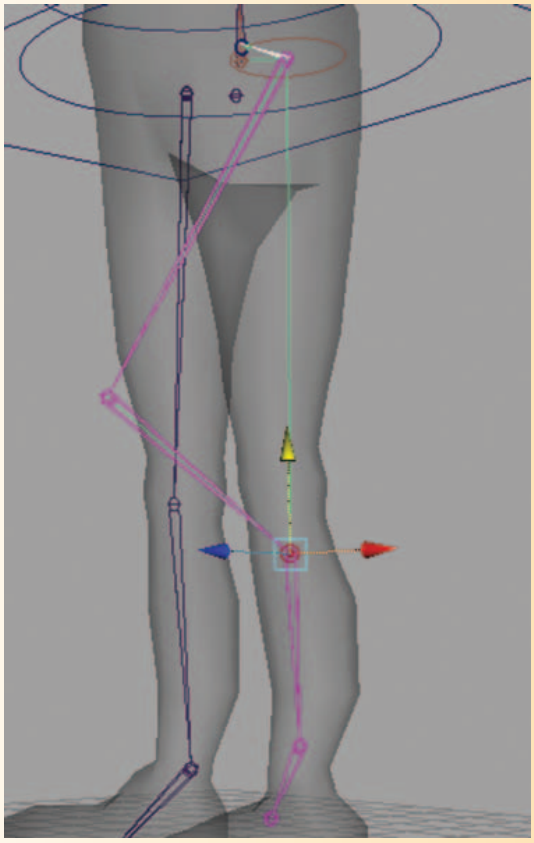
      Place a check mark in the box next to *Sticky*.

      Then **click** "close".
   c. In the PERSPECTIVE window, **click** on the *left_hip* joint (to define the start of the IK joint chain) then on the *left_ankle* joint (to define the end of the chain; an IK handle appears at the end of the chain.) See image on page 251
   d. In the OUTLINER, double-**click** on *ikHandle1* and rename it *leftLeg_ikHandle*. (This chain will control the leg movement.)
   e. It is a great idea to check to make sure the IK handle moves correctly along the Y axis. You can do this by doing the following:
   f. **Select** the *leftLeg_ikHandle* and move the IK handle along the Y axis **(green arrow)** toward the body to confirm that the leg bends in the correct direction. Be sure to press the **(z)** key to undo the move.



Creating an RP IK solver in the left leg.

! The direction that the IK handle points is not relevant. Its direction does not have any effect on the solver.



Moving the leftLeg_ikHandle along the Y axis to make sure the leg bends correctly.

g. Go to **[Skeleton > IK Handle Tool – option box]** and set the following:

   i. Under *IK Handle Settings* change the following:

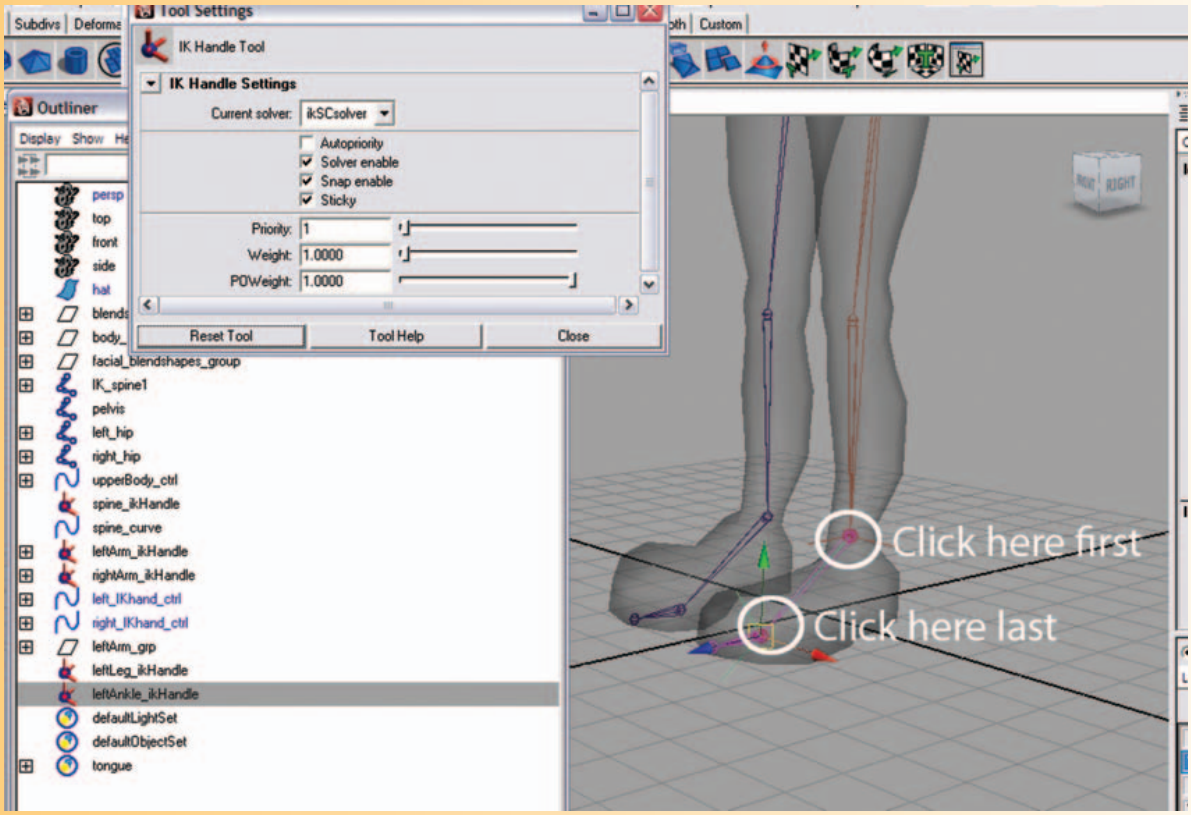
      Current solver: choose "ikSCsolver".

      Keep a check mark in the box next to *Sticky*.

      Then **click** "close".

h. In the PERSPECTIVE window, **click** on the *left_ankle* joint (to define the start of the IK joint chain) then on the *left_ball* joint (to define the end of the chain. An IK handle appears at the end of the chain).

i. In the OUTLINER, double-**click** on *ikHandle1* and rename it *leftAnkle_ikHandle*. (This chain will control the ankle movement.)

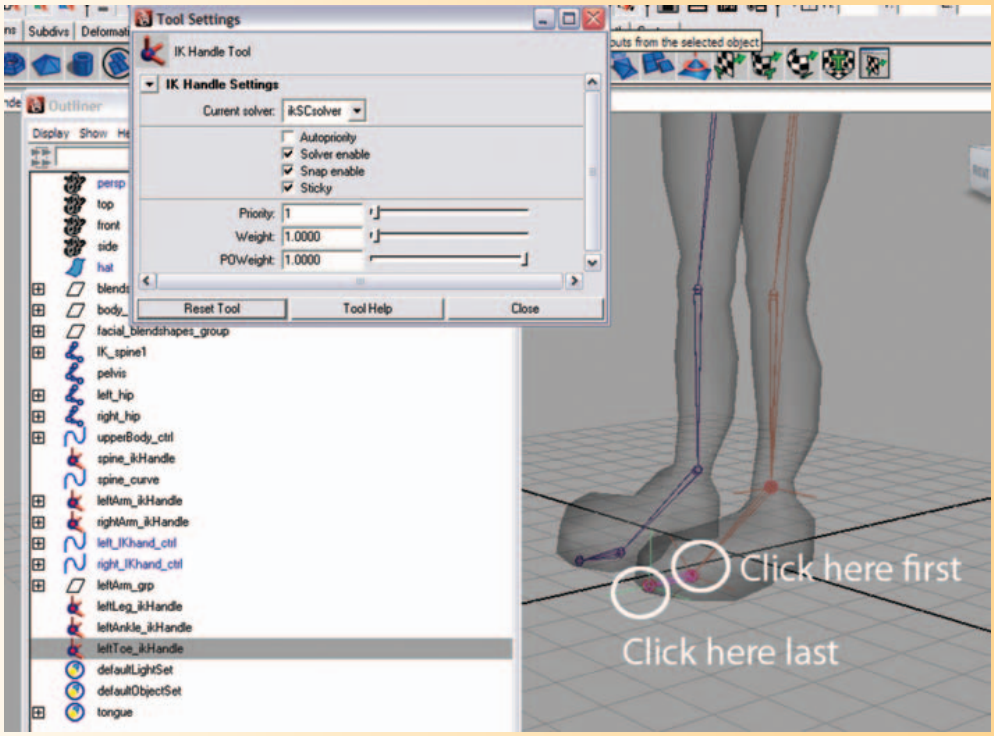


Creating an SC IK solver in the left ankle.

j. Press the **(y)** key to **select** the last tool used – in this case, the IK Handle Tool.

k. In the PERSPECTIVE window, **click** on the *left_ball* joint (to define the start of the IK joint chain) then on the *left_toe* joint (to define the end of the chain; an IK handle appears at the end of the chain).

l. In the OUTLINER, double-**click** on *ikHandle1* and rename it *leftToe_ikHandle*. (This chain will control the toe movement.)



Creating an SC IK solver in the left toe.

m. Repeat steps to create the IK for the right leg.

7. Create a control system for the IK leg by doing the following:

   a. First create the controllers by doing the following:

      i. Go to **[Create > NURBS Primitives > Circle]**.

      ii. In the channel box, rename the circle *left_foot_ctrl*.

      iii. In PERSPECTIVE view, **select** the move tool by pressing **(w)** on the keyboard and reposition the curve around the foot. DO NOT make it even with the sole of the foot as the animator will have a hard time seeing it through the floor plane. Make sure the pivot of the controller is in the ankle by doing the following: With the curve and the move tool still selected, press the (insert) key on the keyboard, hold down the **(v)** key, then position your cursor over the *IK_left_ankle* joint, **click** the **MMB**, and drag it slightly to snap the *left_foot_ctrl* pivot into place in the ankle.

      iv. Use the scale tool by pressing **(r)** and resize the circle if necessary. (This control should be scaled large enough that it is OUTSIDE of the character's foot to make it easy to select.)

      v. With the *left_foot_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

      vi. **Select** the *left_foot_ctrl* and reshape the curve around the foot by doing the following:

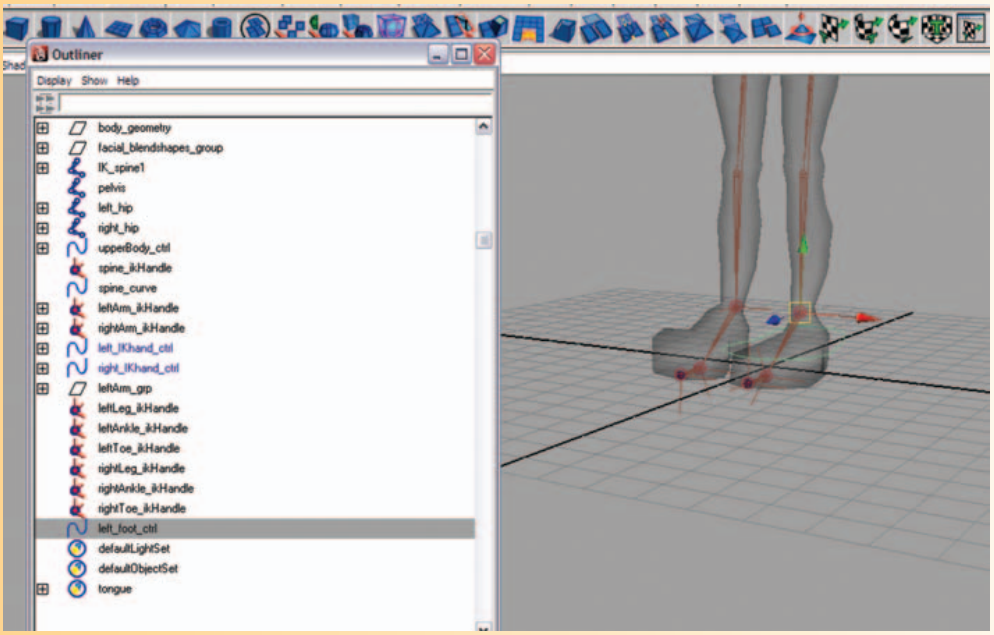
         1. Press the **(F8)** key.

         2. Choose the "**select** point components" button in the Status Line.

3. Using the move tool by pressing **(w)**, **click** and drag around the points of the circle and position them to reshape the curve into a shoeprint shape.

vii. Change the rotation order for the *left_foot_ctrl* by doing the following:

1. With the *left_foot_ctrl* selected, open the attribute editor by pressing **(ctrl+a)**.
2. **Select** the *left_foot_ctrl* tab.
3. Under *Transform Attributes* set the following:
   a. Rotate order: choose "ZXY".



Creating and positioning the left_foot_ctrl.

viii. Duplicate the *left_foot_ctrl* by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

ix. In the OUTLINER, **double-click** on *left_foot_ctrl1* and rename it *right_foot_ctrl*.

x. In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *IK_right_ankle* joint, and **click** the **MMB** and drag it slightly to snap the *right_IKfoot_ctrl* into place.

xi. With the *right_foot_ctrl* selected set the following in the channel box: ScaleX: type "−1" (this will flip the controller over the right foot).

xii. With the *right_foot_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

xiii. Go to **[Create > NURBS Primitives > Circle]**.

xiv. In the channel box, rename the circle *left_knee_ctrl*.

xv. In PERSPECTIVE view, with the *left_knee_ctrl* selected, **select** the move tool by pressing **(w)**, hold down the **(v)** key, position your cursor over the *left*

knee joint, and **click** the **MMB** and drag it slightly to snap the left_*knee_ctrl* into place.

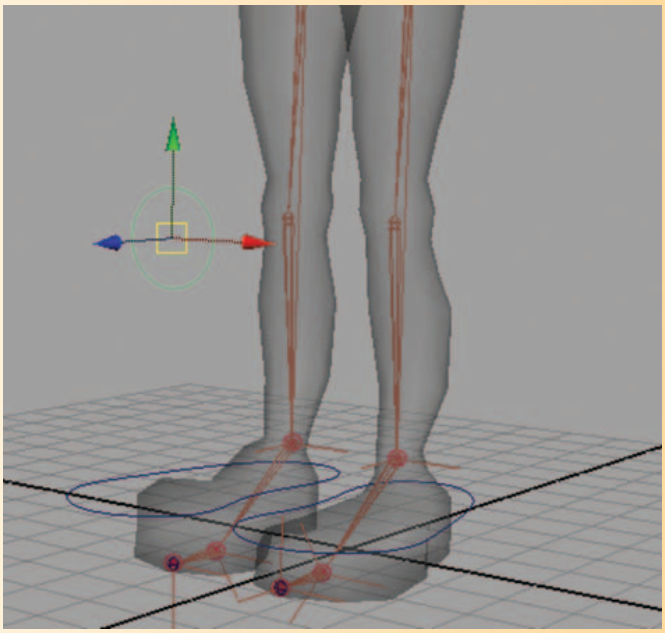
xvi. With the move tool, **click** on the Z axis **(blue arrow)** and move the controller leg distance in front of the character.

xvii. With the *left_knee_ctrl* selected set the following in the channel box: RotateX: type "90".

xviii. Use the scale tool by pressing **(r)** and resize the circle if necessary.

xix. With the *left_knee_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

xx. The rotation order does not need to be changed on this controller, because rotations are not necessary for control.



Creating and positioning the left_knee_ctrl to control the leftLeg_ikHandle's pole vector.

xxi. Duplicate the *left_knee_ctrl* by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

xxii. In the OUTLINER, **double-click** on *left_knee_ctrl1* and rename it *right_knee_ctrl*.

xxiii. In PERSPECTIVE view, **select** the move tool by pressing **(w)** and **click** on the X axis **(red arrow)**, hold down the **(v)** key, position your cursor over the *right_knee* joint, and **click** the LMB and drag it slightly to snap the *right_knee_ctrl* into place. (By **select**ing the X axis first, the move is constrained to that axis only.)

xxiv. With the *right_knee_ctrl* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

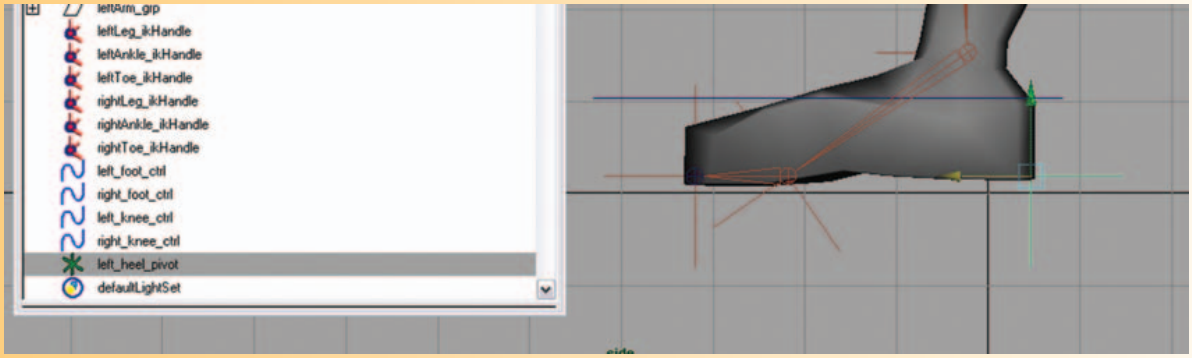
8. You may notice that we have created three separate IK chains for the leg and foot. A biped is capable of moving their toes, their ankle, and their leg independently from each other. Because of this, a single IK chain from the hip to the toe would not work for the control needed. We are now going to create a hierarchal system that provides maximum control and protect the animator from losing their work if an IK handle stops solving. Create control between the controllers and the IK handles by doing the following:

   a. Go to **[Create > Locator]**.

      i. In the channel box, rename the locator *left_heel_pivot*.

      ii. In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, then position your cursor over the *left_ball* joint (the joint in the ball of the foot), **click** the **MMB** and drag it slightly to snap the *left_heel_pivot* into place. **Click** on the Z axis **(blue arrow)** and move the locator to the heel of your character's foot or shoe geometry.

      iii. With the *left_heel_pivot* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)



Creating and positioning a locator in the heel of the foot.

   b. Repeat this again. Go to **[Create > Locator]**.

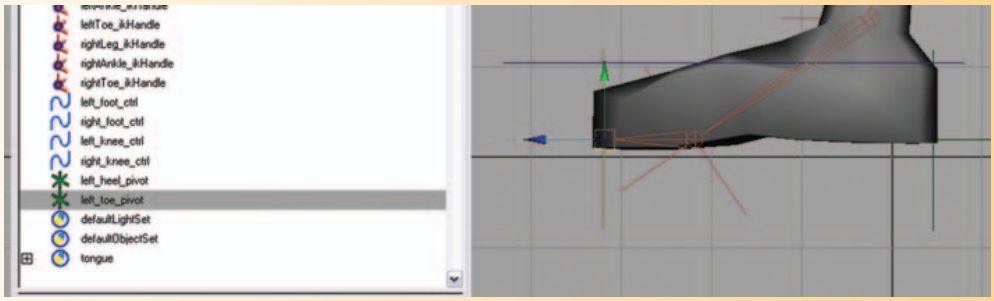
      i. In the channel box, rename the locator *left_toe_pivot*.

      ii. In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, then position your cursor over the *left_toe*, joint, **click** the **MMB** and drag it slightly to snap the *left_toe_pivot* into place.

      iii. With the *left_toe_pivot* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

   c. Repeat this again. Go to **[Create > Locator]**.

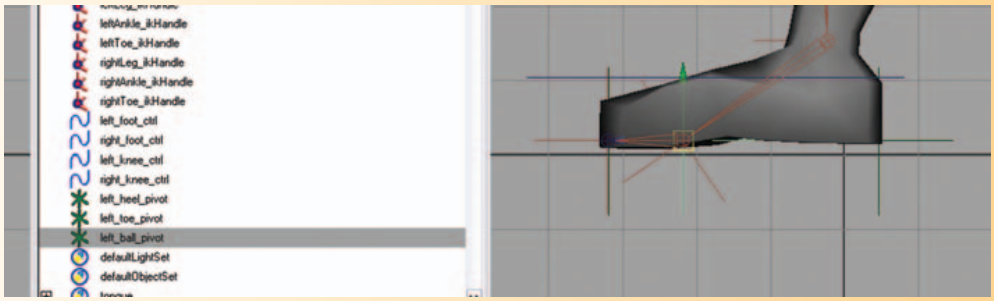
      i. In the channel box, rename the locator *left_ball_pivot*.



Creating and positioning a locator in the toe of the foot.

ii. In PERSPECTIVE view, **select** the move tool by pressing **(w)**, hold down the **(v)** key, then position your cursor over the *left_ball* joint, **click** the **MMB** and drag it slightly to snap the *left_ball_pivot* into place.

iii. With the *left_ball_pivot* selected, go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)



Creating and positioning a locator in the ball of the foot.

iv. Duplicate the *left_ball_pivot* locator by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

v. In the OUTLINER, **double-click** on *left_ball_pivot1* and rename it *left_toe_wiggle*.

d. Repeat (or duplicate) for the right side.

e. In the OUTLINER, **click** on the *leftToe_ikHandle* with the **MMB** and drag it onto the *left_toe_wiggle* locator. (This makes the *leftToe_ikHandle* child to the *left_toe_wiggle* locator.)

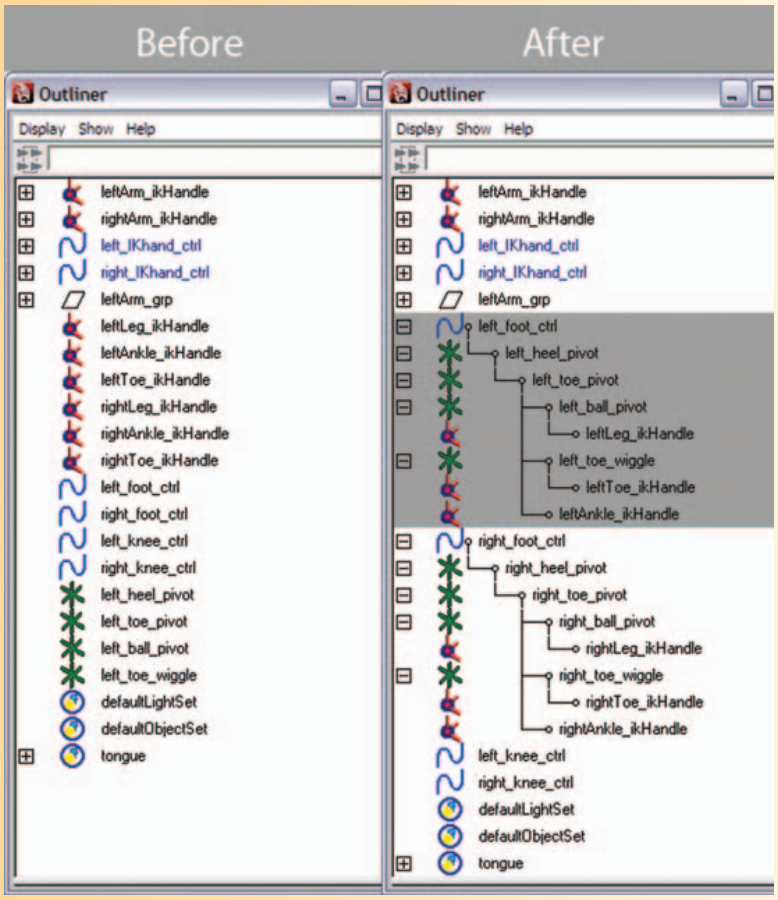
f. In the OUTLINER, **click** on the *leftLeg_ikHandle* with the **MMB** and drag it onto the *left_ball_pivot* locator. (This makes the *leftLeg_ikHandle* child to the *left_ball_pivot* locator.)

g. In the OUTLINER, **click** on the *left_ball_pivot* locator, hold down the **(ctrl)** key on the keyboard and **click** on the *left_toe_wiggle* locator, **click** on the

leftAnkle_ikHandle, **click** on the *left_toe_pivot* locator and then press **(p)** on the keyboard. (This makes the *left_ball_pivot* locator, the *left_toe_wiggle* locator, and the *leftAnkle_ikHandle* child to the *left_toe_pivot* locator.)

h. In the OUTLINER, **click** on the *left_toe_pivot* locator with the **MMB** and drag it onto the *left_heel_pivot* locator. (This makes the *left_toe_pivot* locator child to the *left_heel_pivot* locator.)

i. In the OUTLINER, **click** on the *left_heel_pivot* locator with the **MMB** and drag it onto the *left_foot_ctrl*. (This makes the *left_heel_pivot* locator child to the *left_foot_ctrl*.)

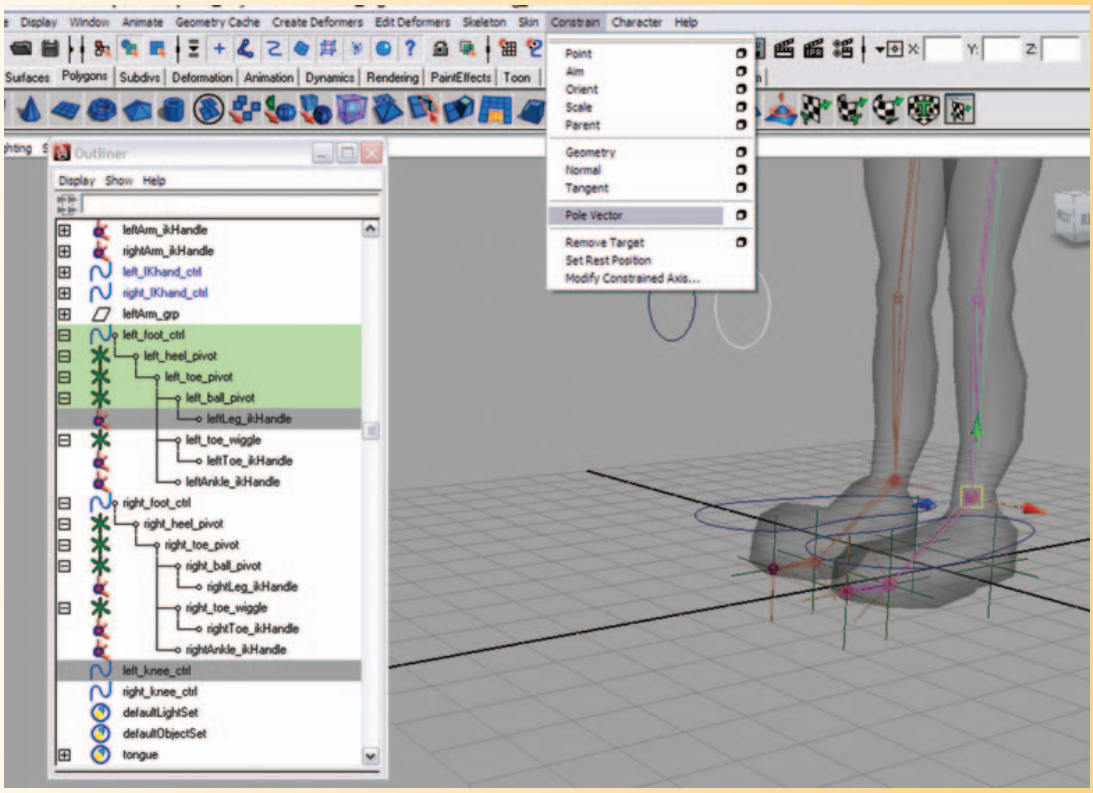


The OUTLINER hierarchy before and after parenting the IK handles with the locators to make the foot hierarchy.

j. In the PERSPECTIVE window, **click** on the *left_knee_ctrl* (the leader), hold down the **(ctrl)** key and **click** on the *leftLeg_ikHandle* (the follower), then go to **[Constrain > Pole Vector]**.

k. In the OUTLINER, **click** on the *left_IKknee_ctrl* with the **MMB** and drag it onto the *left_foot_ctrl*. (This makes the *left_IKknee_ctrl* child to the *left_ foot_ctrl*.)

l. Repeat for the right leg.



Creating a pole vector constraint between the *left_knee_ctrl* (leader) and the *leftLeg_ikHandle* (follower).

You can now test out some of your leg controls. **Select** the *left_foot_ctrl* and move it around toward the body so that the knee bends. You can also rotate this control to control the ankle. **Select** the *left_knee_ctrl* and move it left to right to control the position of the knee. Be sure to press the **(z)** key several times to undo the moves.

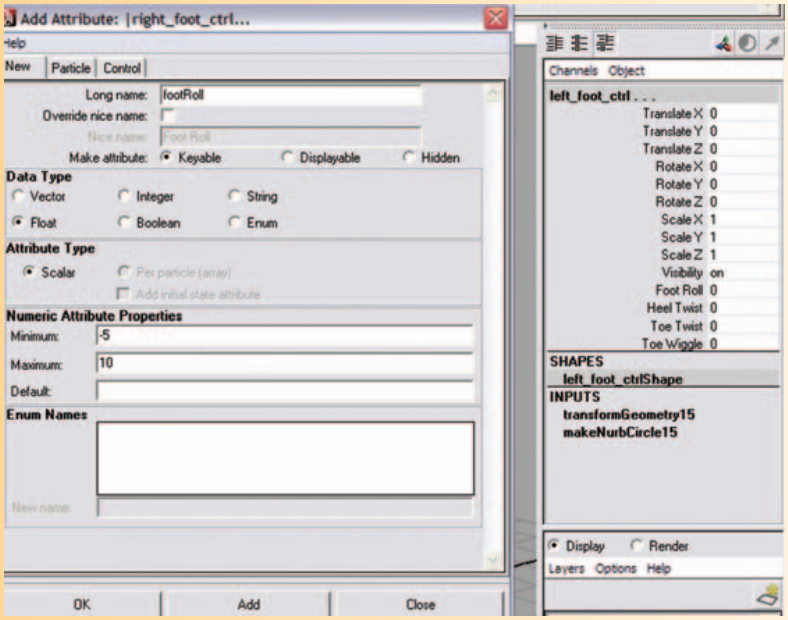
9. Add attributes to the foot control for the foot movements by doing the following:
    a. **Select** the *right_foot_ctrl*, hold down the **(shift)** key and **select** *left_foot_ctrl*, then go to **[Modify > Add Attribute]** and **enter** the following:
        i. Attribute name: type "footRoll".
        ii. Under *Numeric Attribute Properties*
            1. Minimum: type "−5".
            2. Maximum: type "**10**".
            3. **Click** "**Add**".
        iii. Attribute name: type "**heelTwist**"
            1. **Click** "**Add**".

iv. Attribute name: type "**toeTwist**"

1. **Click** "**Add**".

v. Attribute name: type "**toeWiggle**"

1. **Click** "**OK**".



Adding custom attributes to the left_foot_ctrl and right_foot_ctrl using [Modify > Add Attribute].

10. Make the footRoll attribute function using *Set Driven Key* by doing the following:

a. In the OUTLINER, **select** the *left_heel_pivot* locator, hold down the **(ctrl)** key and **click** the *left_toe_pivot* locator and the *left_ball_pivot* locator, then go to **[Animate > Set Driven Key > Set...]**.

! When you are using Set Driven Key, remember to change the driver first, then the driven, then set a key (you are changing the pose for each keyframe, much like you do in the timeline when animating, but this is an attribute value instead of a time position).

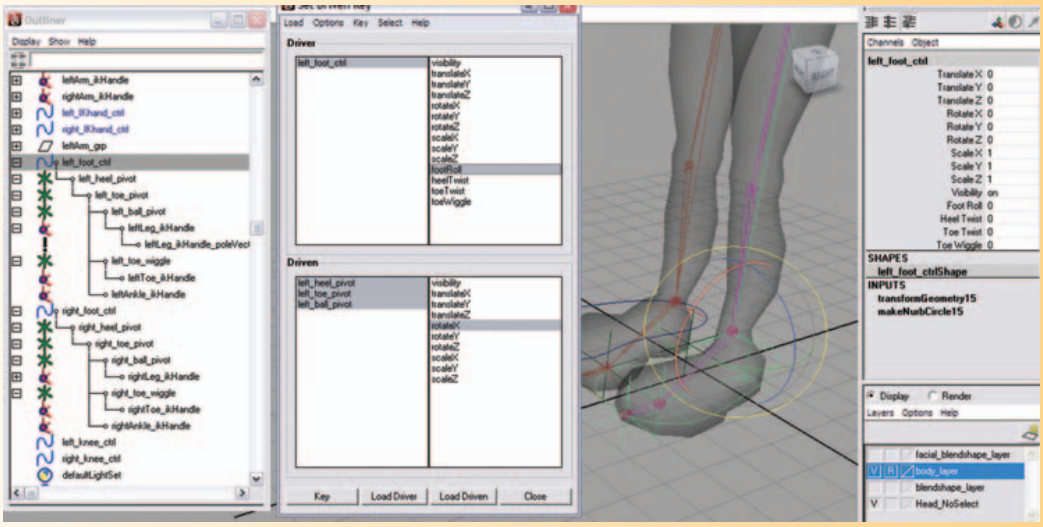
b. **Select** the *left_foot_ctrl* and **click** "Load Driver" in the *Set Driven Key* window.

i. In the *Driver* section of the *Set Driven Key* window, choose "footRoll" in the right column.

ii. In the *Driven* section of the *Set Driven Key* window, **click** on *the left_heel_pivot*, hold down the **(ctrl)** key and **click** the *left_toe_pivot* and *left_ball_pivot* to **select** them.

iii. In the *Driven* section of the *Set Driven Key* window, choose "RotateX" in the right column.

iv. In the *Set Driven Key* window, **click** "**Key**". (This sets a default position of the foot at the footRoll value of "**0**".)



Loading the Set Driven Key window and setting the first key so that when the footRoll attribute is set to "0", the foot is in the default (original) position.

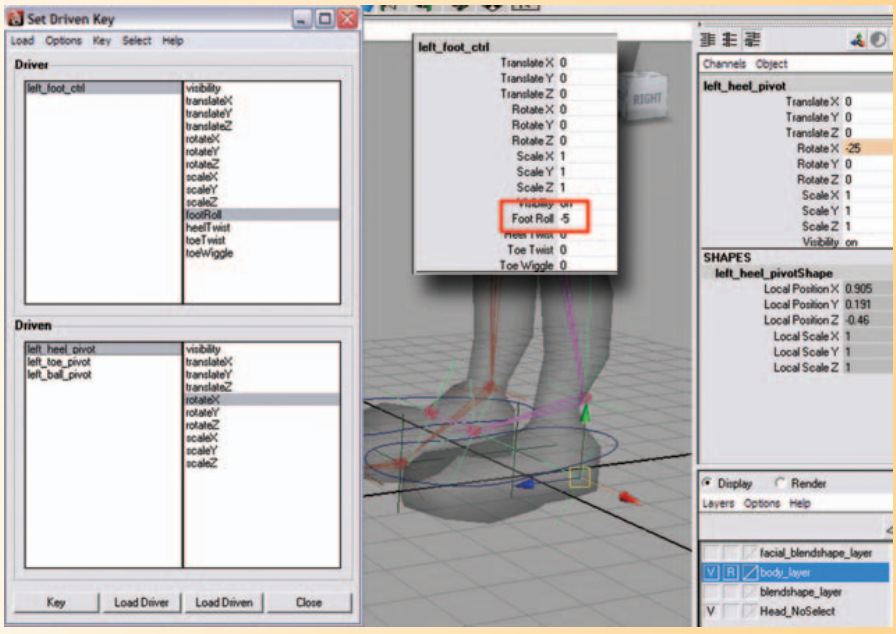
v. In the *Driver* section of the *Set Driven Key* window, **click** on *left_foot_ctrl* to **select** it.

vi. In the channel box, change *left_foot_ctrl* to "−5".

vii. In the *Driven* section of the *Set Driven Key* window, **click** on *left_heel_pivot* to **select** it.

viii. In the channel box, change *RotateX* to "−25".

ix. In the *Set Driven Key* window, **click** "**Key**". (This sets the key for the first pose of the foot roll – the heel contacting the ground with the toe raised.)



Setting the second key so that when the footRoll is set to "−5", the heel is planted on the ground and the toe is in the air.

x. In the *Driver* section of the *Set Driven Key* window, **click** on *left_foot_ctrl* to **select** it.

xi. In the channel box, change *left_foot_ctrl* to "5".

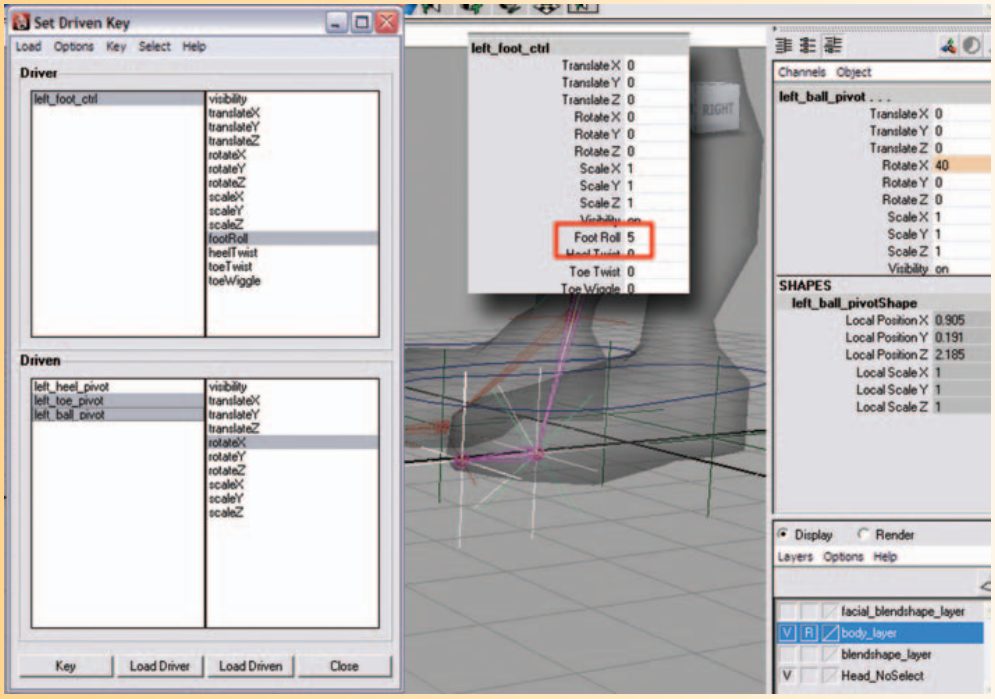
xii. In the *Driven* section of the *Set Driven Key* window, **click** on *left_ball_pivot* to **select** it.

xiii. In the channel box, change *RotateX* to "40".

xiv. In the *Set Driven Key* window, **click** "**Key**". (The second pose is already keyed at the default position, so this is the third pose of the foot roll – the heel leaving the ground with the ball on the ground.)

xv. In the *Driven* section of the *Set Driven Key* window, **click** on *left_toe_pivot* to **select** it.

xvi. In the *Set Driven Key* window, **click** "**Key**".



Setting the third key so that when the footRoll is set to "5", the ball is planted on the ground and the heel is in the air.

xvii. In the *Driver* section of the *Set Driven Key* window, **click** on *left_foot_ctrl* to **select** it.

xviii. In the channel box, change *left_foot_ctrl* to "**10**".

xix. In the *Driven* section of the *Set Driven Key* window, **click** on *left_toe_pivot* to **select** it.

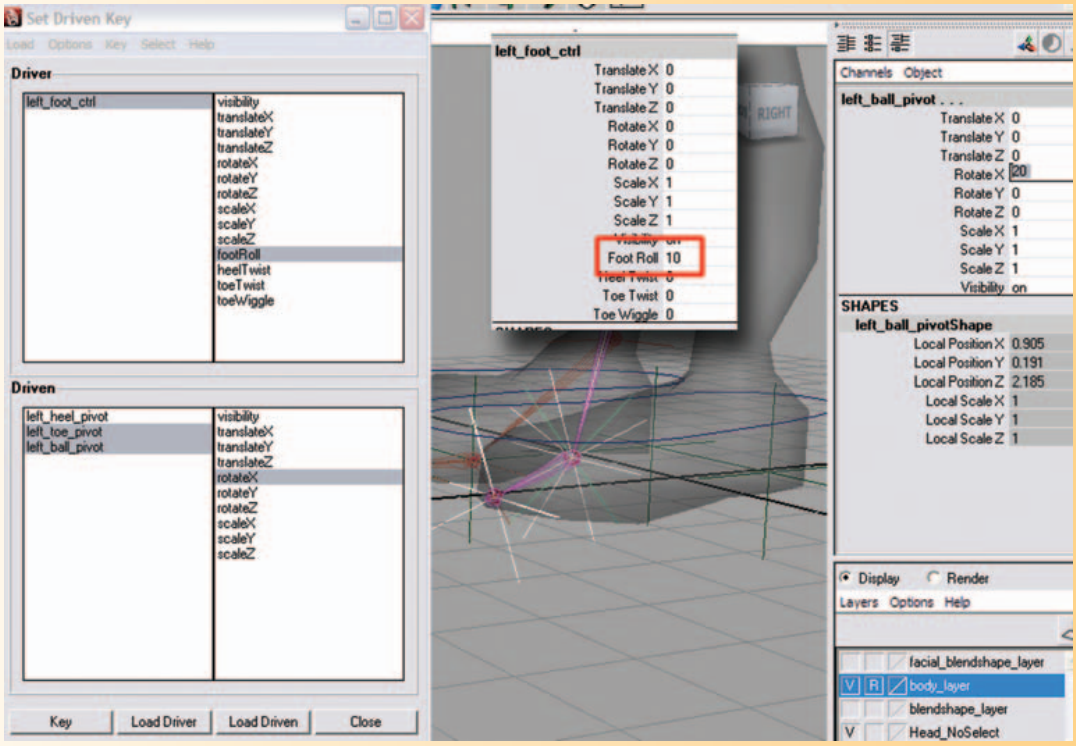
xx. In the channel box, change *RotateX* to "20".

xxi. In the *Set Driven Key* window, **click** "**Key**".

xxii. In the *Driven* section of the *Set Driven Key* window, **click** on *left_ball_pivot* to **select** it.

xxiii. In the channel box, change *RotateX* to "20".

xxiv. In the *Set Driven Key* window, **click** "**Key**". (This sets the key for the fourth pose of the foot roll – the ball leaving the ground with the tip of the toe on the ground.)



Setting the fourth key so that when the footRoll is set to "10", the ball lifts off of the ground with the heel in the air.

c. Test the foot roll to make sure that it works properly.

i. In the *Driver* section of the *Set Driven Key* window, **click** on *left_foot_ctrl* to **select** it.

ii. In the channel box, **click** on the words *Foot Roll*.

iii. Place your cursor in the PERSPECTIVE window, **click** and hold your **MMB**, and drag left to right to see the foot roll.

d. Repeat for the right leg.

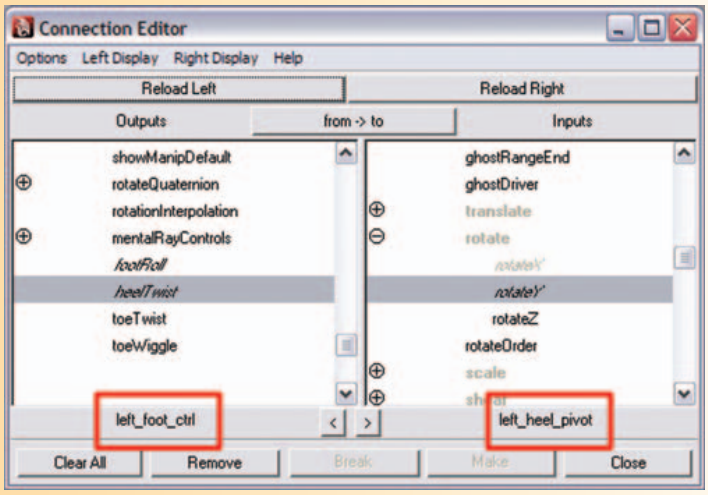
! Remember, if you make a mistake you can start all over again by breaking the connections in the channel box. In the OUTLINER, **select** the *left_heel_pivot* locator, hold down the **(ctrl)** key and **click** the *left_toe_pivot* locator and the *left_ball_pivot* locator, then **click** on the word RotateX in the channel box, RMB and choose "break connections" which will delete the keyframes and you can start again.

11. Make the remaining attributes function using the *Connection Editor*. Do the following:

    a. Go to **[Window > General Editors > Connection Editor]**.

    b. In the OUTLINER, **select** the *left_foot_ctrl* and **click** "Reload Left" Scroll down to the bottom of the list and choose "heelTwist".

    c. In the OUTLINER, **select** the *left_heel_pivot* and **click** "Reload Right" Scroll down to Rotate and **click** on the **(+)** to open the values. Choose "RotateY" to make the connection.

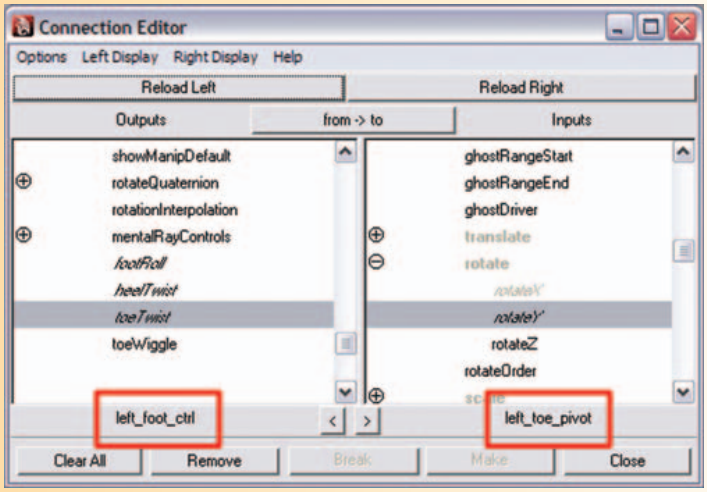


Using the connection editor to create a direct relationship between the left_foot_ctrl.heelTwist attribute and the left_heel_Pivot.rotateY attribute.

    d. **Click** "Clear All".

    e. In the OUTLINER, **select** the *left_foot_ctrl* and **click** "Reload Left" Scroll down to the bottom of the list and choose "toeTwist".

    f. In the OUTLINER, **select** the *left_toe_pivot* and **click** "Reload Right" Scroll down to Rotate and **click** on the **(+)** to open the values. Choose "RotateY" to make the connection.

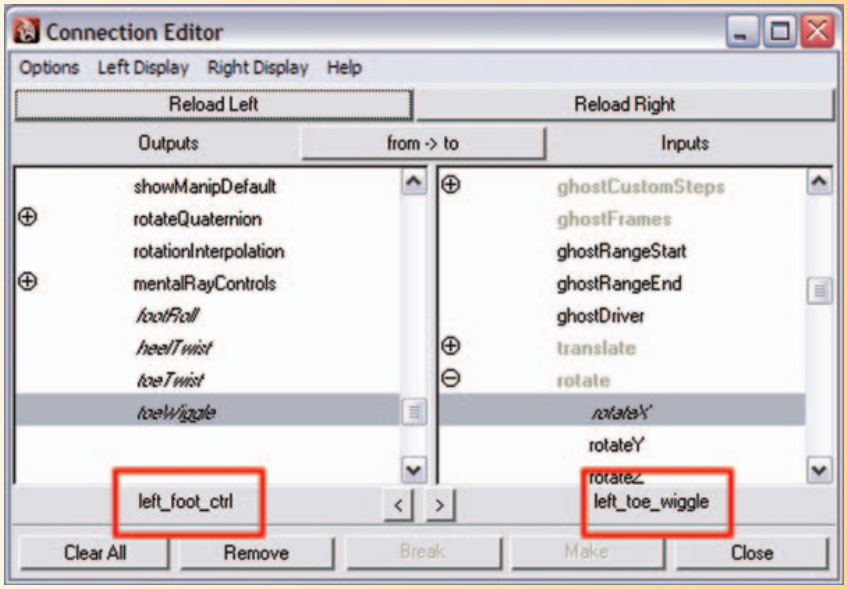


Using the connection editor to create a direct relationship between the left_foot_ctrl.toePivot attribute and the left_toe_Pivot.rotateY attribute.

g. **Click** "Clear All".

h. In the OUTLINER, **select** the *left_foot_ctrl* and **click** "Reload Left" Scroll down to the bottom of the list and choose "toeWiggle".

i. In the OUTLINER, **select** the *left_toe_wiggle* and **click** "Reload Right" Scroll down to Rotate and **click** on the **(+)** to open the values. Choose "RotateX" to make the connection.



Using the connection editor to create a direct relationship between the left_foot_ctrl.toeWiggle attribute and the left_toe_wiggle.rotateX attribute.

j. Test the attributes to make sure that they work properly.

i. In the OUTLINER section **click** on *left_foot_ctrl* to **select** it.

ii. In the channel box, **click** on the words *heelTwist*.

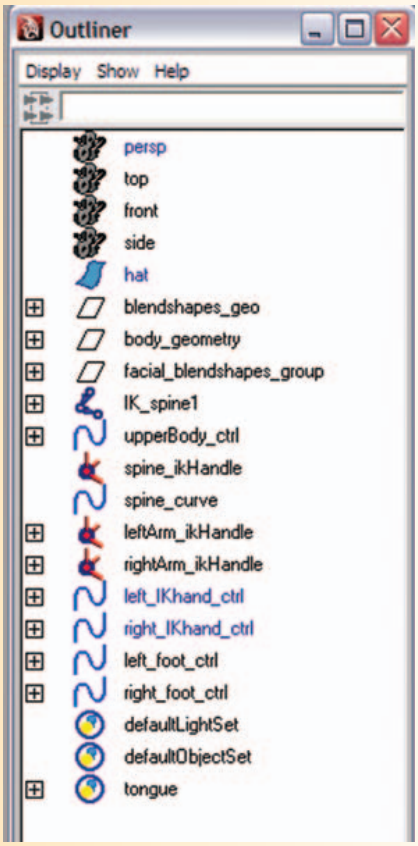
iii Place your cursor in the PERSPECTIVE window, **click** and hold your **MMB**, and drag left to right to see the foot roll, repeat the test for *toeTwist* and *toeWiggle*.

k. Repeat the connections for the right leg.

12. Integrate the legs into the existing spine controls by doing the following:

a. In the OUTLINER, **select** the *left_hip*, hold down the **(ctrl)** key, **click** the *right_hip*, the *pelvis*, and press **(p)** to parent.

b. In the OUTLINER, **select** the *pelvis*, hold down the **(shift)** key, and in the PERSPECTIVE window **click** the *hips_ctrl* and press **(p)** to parent.

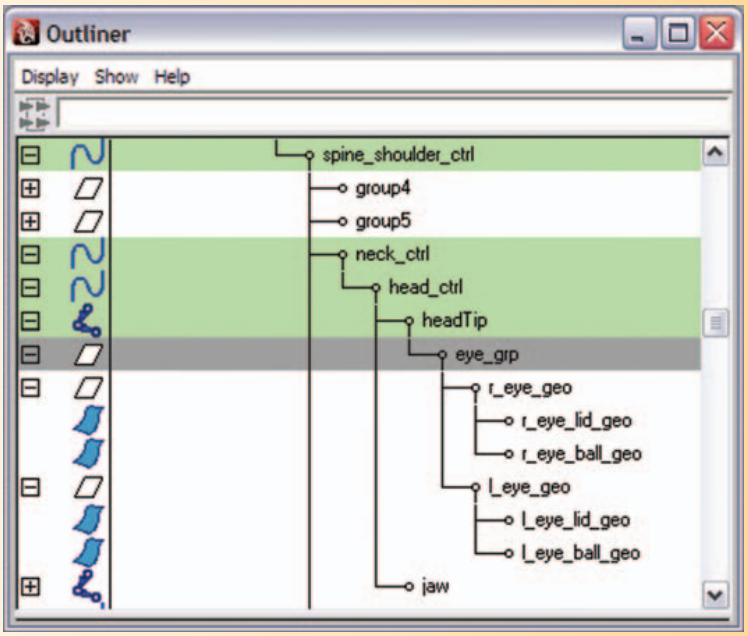


The new OUTLINER hierarchy after integrating the legs into the existing spine controls.

13. Save your scene file. Name your scene *06_asgn06.ma*.

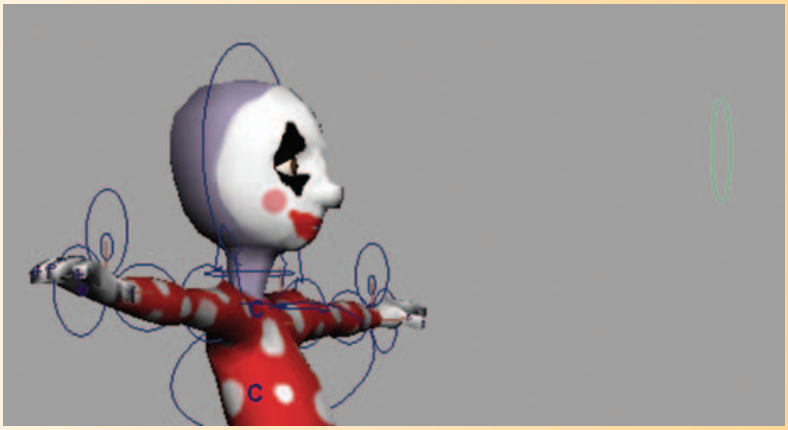
Assignment 6.7: Creating a Control System for the Eyes and Face

1. Open Maya and set your project.
   a. From your computer's desktop, go to **[Start > Programs]** and **select** Maya.
   b. Once Maya is open go to **[File > Project > Set...]** and browse to your project folder then **click** "**OK**".
2. Open your last saved file: Go to **[File > Open]** and **select** *06_asgn06.ma*.
3. Turn X-ray Joints mode off by going to **[Shading > X-ray Joints]**.
4. Make sure that your geometry layer is set to **R** for reference so that you are unable to **select** the geometry by mistake when working.
5. To make selection easier open your OUTLINER by going to **[Windows > Outliner]**.
6. The eyes become part of the skeletal structure so that they move along with the skeleton, but do not deform. In the OUTLINER, **select** the *right_eye_geo* and *left_eye_geo*, then group them together by pressing **(ctrl+g)**. Double-**click** on the group in the OUTLINER and rename the group *eye_geo_group*.
7. In the PERSPECTIVE view, with the *eye_geo_group* selected, shift **select** the *headTip* joint and press **(p)** to parent the *eye_geo_group* to the top joint in the head.



Parenting the Eye groups to the *headTip joint*.

8. Create a control system for the eyes by doing the following:
   a. Go to **[Create > NURBS Primitives > Circle]**.
   b. In the channel box, rename *nurbsCircle1* to *eye_ctrl*.
   c. With the *eye_ctrl* selected set the following in the channel box: RotateX: type "90".
   d. In the FRONT orthographic with the *eye_ctrl* selected, **select** the move tool by pressing **(w)** and position the *eye_ctrl* in front of your character's face.
   e. In the PERSPECTIVE view, with the move tool, **click** on the Z axis **(blue arrow)** and move the controller in front of your character's face, about half an arm's length.
   f. With the *eye_ctrl* selected, **select** the scale tool by pressing **(r)** and scale if necessary.



Creating and positioning the *eye_ctrl*.

g. Duplicate the *eye_ctrl* by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

h. In the channel box, rename *eye_ctrl1* to *left_eye_ctrl*.

i. With the *left_eye_ctrl* selected, **select** the scale tool by pressing **(r)** and scale it smaller.

j. In the FRONT orthographic view, with the *left_eye_ctrl* selected, **select** the move tool by pressing **(w)** and position the *left_eye_ctrl* in front of the *left_eye*.

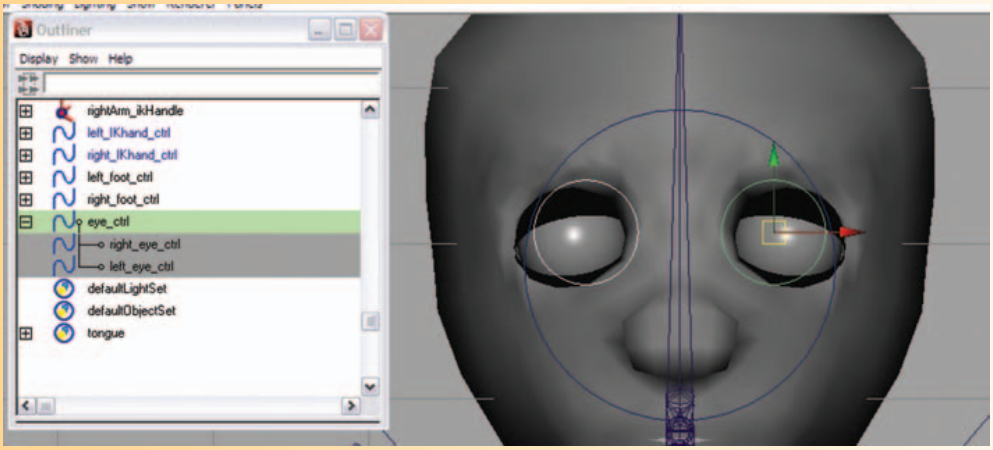
k. Duplicate the *left_eye_ctrl* by going to **[Edit > Duplicate]** or press **(ctrl+d)**.

l. In the OUTLINER, **double-click** on *left_eye_ctrl1* and rename it *right_eye_ctrl*.

m. In PERSPECTIVE view, **select** the move tool by pressing **(w)** and **click** on the X axis **(red arrow)**, and position the *right_eye_ctrl* in front of the *right_eye*.

n. With the *right_eye_ctrl* selected, hold down the **(shift)** key and **select** *left_eye_ctrl* and *eye_ctrl* then go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)

o. The rotation order does not need to be changed on these controllers, because rotations are not necessary for control.



Creating and positioning the *left_eye_ctrl* and the *right_eye_ctrl*.

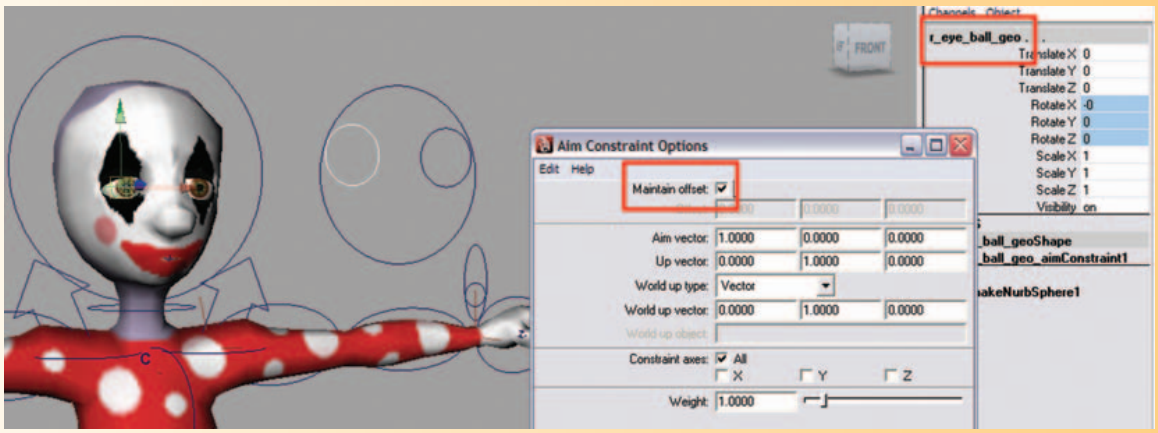
p. In the OUTLINER, **click** on the *left_eye_ctrl* with the **MMB** and drag it onto the *eye_ctrl*. (This makes the *left_eye_ctrl* child to the *eye_ctrl*.)

q. In the OUTLINER, **click** on the *right_eye_ctrl* with the **MMB** and drag it onto the *eye_ctrl*. (This makes the *right_eye_ctrl* child to the *eye_ctrl*.)

r. In the OUTLINER, **click** on the *eye_ctrl* with the **MMB** and drag it onto the *upperBody_ctrl*. (This makes the *eye_ctrl* child to the *upperBody_ctrl*.)

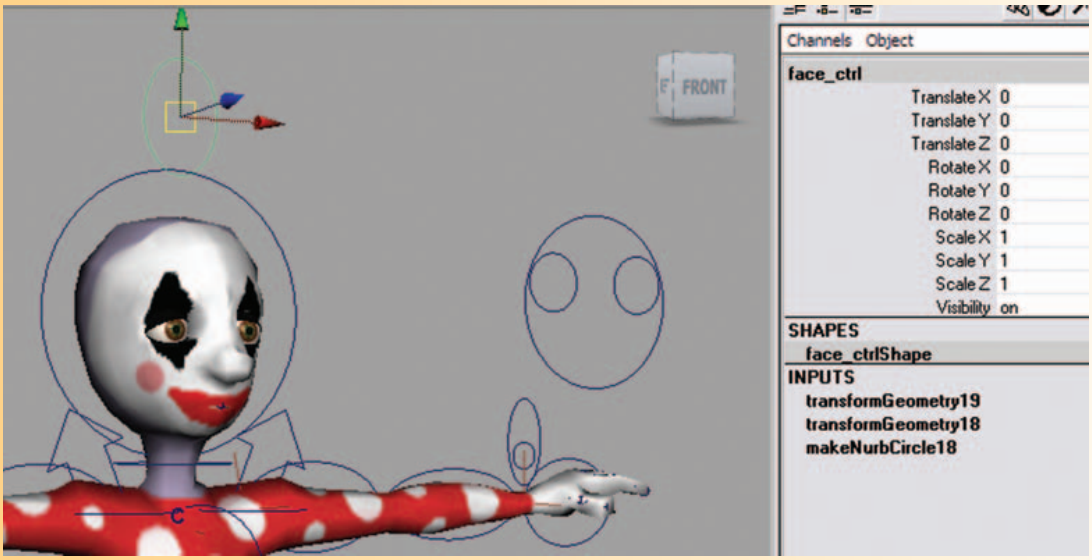
s. In the PERSPECTIVE window, **select** the *eye_ctrl*, hold down the **(shift)** key, **click** the *upperBody_ctrl* and press **(p)** to parent. (Parenting the *eye_ctrl* to the *upperBody_ctrl* provides the ability to fix the character's gaze in space. As another option, you could parent the *eye_ctrl* to the *face_ctrl* which is created next.)

9. Make the controllers function using *Constraints* by doing the following:
   a. In the PERSPECTIVE window, **click** on the *left_eye_ctrl* (the leader), hold down the **(ctrl)** key, and **click** on the left *eyeball_geo* (the follower), then go to **[Constrain > Aim – option box]**.
   b. Place a check mark in the box next to *Maintain Offset*.
   c. Then **click** "**add**".
   d. Repeat this for the right eye.
   e. You can test the controls out by selecting them and moving them around in the PERSPECTIVE window.



Creating an aim constraint between the *left_eye_ctrl* (leader) and the left eyeball geometry (follower). Make sure the geometry is selected second to add the constraint correctly.

10. Create a control system for the facial expression by doing the following:
   a. Go to **[Create > NURBS Primitives > Circle]**.
   b. In the channel box, rename *nurbsCircle1* to *face_ctrl*.
   c. With the *face_ctrl* selected set the following in the channel box: **RotateX**: type "**90**".
   d. In the PERSPECTIVE view panel, with the *face_ctrl* selected, **select** the move tool by pressing **(w)** and position the *face_ctrl* above your character's head.
   e. With the *face_ctrl* selected, **select** the scale tool by pressing **(r)** and scale if necessary.
   f. The rotation order does not need to be changed on this controller, because rotations are not necessary for control.
   g. In the PERSPECTIVE view, **click** on the *face_ctrl*, hold down the **(shift)** key, click on the *head_ctrl*, and then press **(p)** to parent. (This makes the *face_ctrl* child to the *head_ctrl*.)
   h. With the *face_ctrl* selected go to **[Modify > Freeze Transformations]**. (To return both translate and rotate values to 0 and the scale values to 1.)



Creating and positioning the *left_eye_ctrl* and the *right_eye_ctrl*.

11. Add attributes to the face control to control the blend shapes by doing the following:
    a. **Select** the *face_ctrl* and go to **[Modify > Add Attribute]** and **enter** the following:
        i. Attribute name: type "leftEyeBlink".
        ii. Under *Numeric Attribute Properties*
            1. Minimum: type "−10".
            2. Maximum: type "**10**".
            3. **Click** "**Add**".
        iii. Attribute name: type "rightEyeBlink".
            1. Minimum: type "−10".
            2. Maximum: type "**10**".
            3. **Click** "**Add**".
        iv. Attribute name: type "leftEyebrowRaise"
            1. Minimum: type "−10".
            2. Maximum: type "**10**".
            3. **Click** "**Add**".
        v. Attribute name: type "rightEyebrowRaise"
            1. Minimum: type "−10".
            2. Maximum: type "**10**".
            3. **Click** "**Add**".
        vi Attribute name: type "leftEyebrowFurrow"
            1. Minimum: type "−10".
            2. Maximum: type "**10**".
            3. **Click** "**Add**".

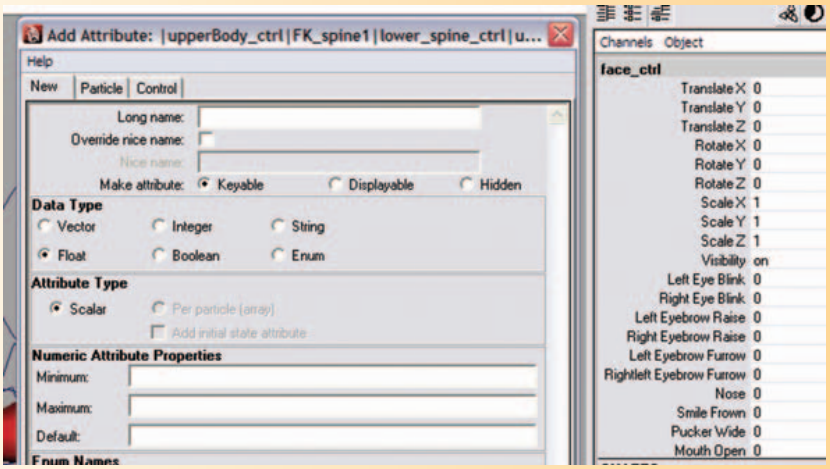
vii. Attribute name: type "rightEyebrowFurrow"
   1. Minimum: type "−10".
   2. Maximum: type "**10**".
   3. **Click** "**Add**".

viii. Attribute name: type "nose"
   1. Minimum: type "−10".
   2. Maximum: type "**10**".
   3. **Click** "**Add**".

ix. Attribute name: type "smileFrown"
   1. Minimum: type "−10".
   2. Maximum: type "**10**".
   3. **Click** "**Add**".

x. Attribute name: type "puckerWide"
   1. Minimum: type "−10".
   2. Maximum: type "**10**".
   3. **Click** "**Add**".

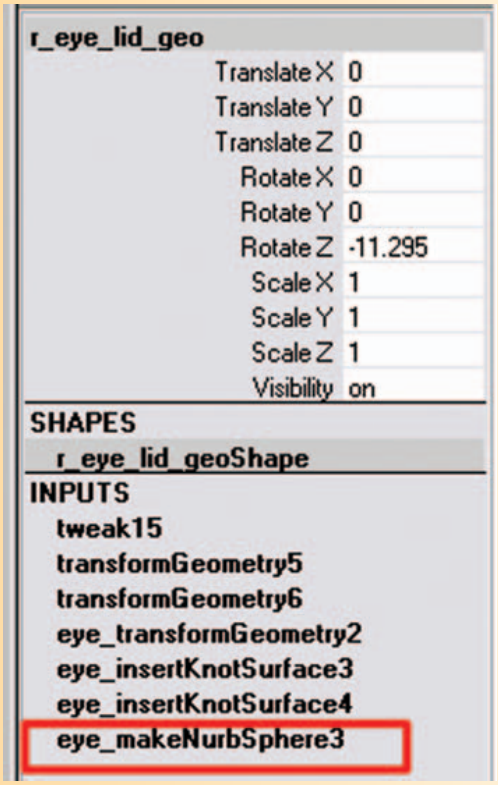
xi. Attribute name: type "mouthOpen"
   1. Minimum: type "**0**".
   2. Maximum: type "**10**".
   3. **Click** "**Add**".



Adding custom attributes to the *face_ctrl* using [Modify > Add Attribute].

12. Before making the controls work, you must import your referenced model (at this point your model MUST be completed before importing). If you have not done so already, do the following:
   a. Go to **[File > Reference Editor]**.
   b. **Click** on the file in the bottom half of the reference editor.
   c. Go to **[File > Import Objects from Reference]**.

13. Delete history on geometry that will be skinned. This should already be done, but it is a good idea to make sure that history is deleted on all skinable geometry. Go to **[Edit > Delete by Type > Non Deformer History]**. DO NOT DELETE HISTORY ON EYELIDS, as this will remove the makeNurbsSphere input, and the eye blinks will no longer work.

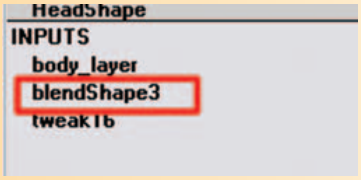


To verify that the INPUTS still exist on the eyelids, click on each eyelid and check the channel box.

14. Make sure blend shapes are applied by clicking on the base shape and checking the INPUT section of the channel box. If the blend shape input is not listed, then do the following:

    a. **Select** your target shapes then shift **select** your base shape.

    b. **Select [Create Deformers > Blend Shapes]**.



Verifying that the blend shape input still exist on the geometry base shape by checking the INPUTS in the channel box.

15. Make the *leftEyeBlink* attribute function using *Set Driven Key* by doing the following:

    a. In the PERSPECTIVE window, **select** the left *eyeball_geo*. In the channel box, under the INPUTS section, **click** on *makeNurbSphere*, then go to **[Animate > Set Driven Key > Set...]**.

    b. **Select** the *face_ctrl* and **click** "Load Driver" in the *Set Driven Key* window.

        i. In the *Driver* section of the *Set Driven Key* window, choose "leftEyeBlink" in the right column.

        ii. In the *Driven* section of the *Set Driven Key* window, **click** on the *makeNurbsSphere* to **select** it.

> **!** If your eyelid does not have the input for *makeNurbSphere1* you will need to recreate the eyes, or find an earlier version of your saved files and import those into this file.

        iii. In the *Driven* section of the *Set Driven Key* window, in the right column choose "startSweep" then hold down the **(ctrl)** key and **click** on "endSweep".

        iv. In the *Set Driven Key* window, **click** "**Key**". (This sets a default position of the leftEyeBlink value of "**0**".)

Loading the Set Driven Key window and setting the first key so that when the *leftEyeBlink* attribute is set to "0", the face is in the default (original) position.

        v. In the *Driver* section of the *Set Driven Key* window, **click** on *face_ctrl* to **select** it.

        vi. In the channel box, change *leftEyeBlink* to "**10**".

        vii. In the *Driven* section of the *Set Driven Key* window, **click** on *makeNurbsSphere* to **select** it.

        viii. Change the start and end sweep to a closed position. In the channel box, change startSweep to about "**0**", change endSweep to about "359.8".

        ix. In the *Set Driven Key* window, **click** "**Key**".

Setting the second key so that when the *leftEyeBlink* is set to "10", the left eye is blinking.

x. In the *Driver* section of the *Set Driven Key* window, **click** on *face_ctrl* to **select** it.

xi. In the channel box, change *leftEyeBlink* to "−10".

xii. In the *Driven* section of the *Set Driven Key* window, **click** on *makeNurbsSphere* to **select** it.

xiii. Change the start and end sweep to a wide eyed position. In the channel box, change startSweep to about "70", change endSweep to about "315" (these values could be more or less for your character).

Setting the third key so that when the *leftEyeBlink* is set to "−10", the left eye is open wide.

xiv. In the *Set Driven Key* window, **click** "**Key**".

c. Test the *leftEyeBlink* to make sure that it works properly.

i. In the *Driver* section of the *Set Driven Key* window, **click** on *face_ctrl* to **select** it.

ii. In the channel box, **click** on the words *Left Eye Blink*. Place your cursor in the PERSPECTIVE window, **click** and hold your **MMB**, and drag left to right to see the eye blink.

d. Repeat for the *rightEyeBlink*.

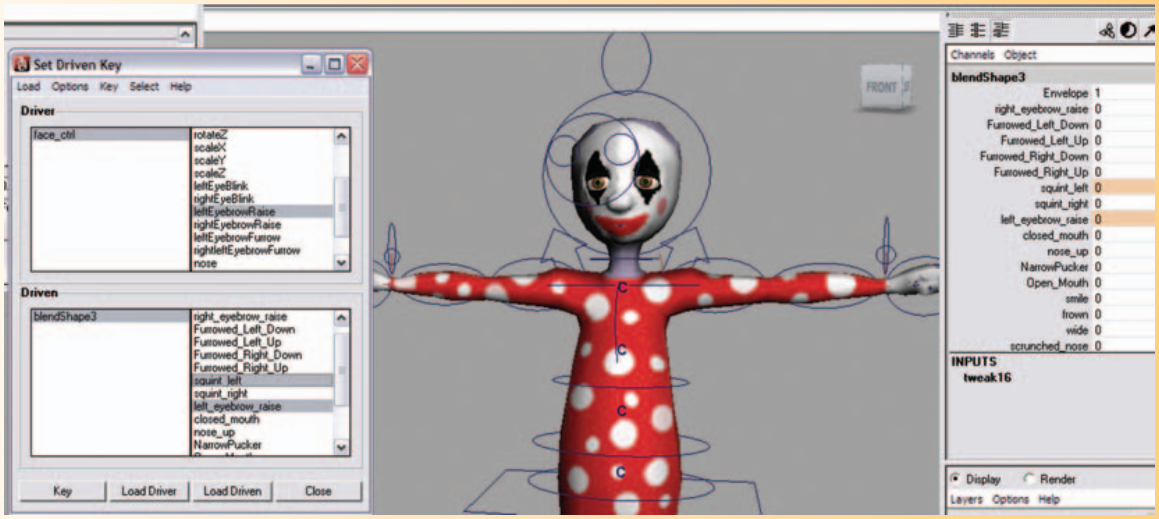
16. Make the *leftEyebrowRaise* attribute function using *Set Driven Key* by doing the following:

a. In the *Driver* section of the *Set Driven Key* window, choose "leftEyebrowRaise" in the right column.

b. Go to **[Window > Animation Editors > Blend Shape] click** on the "**select**" button in the blend shape window. Then **click** on "Load Driven" in the *Set Driven Key* window.

i. In the *Driven* section of the *Set Driven Key* window, in the right column choose "left_eyebrow_raise" then hold down the **(ctrl)** key and **click** on "left_squint".

ii. In the *Set Driven Key* window, **click** "**Key**". (This sets a default position of the *leftEyebrowRaise* value of "**0**".)



Loading the Set Driven Key window and setting the first key so that when the *leftEyebrowRaise* attribute is set to "0", the face is in the default (original) position.

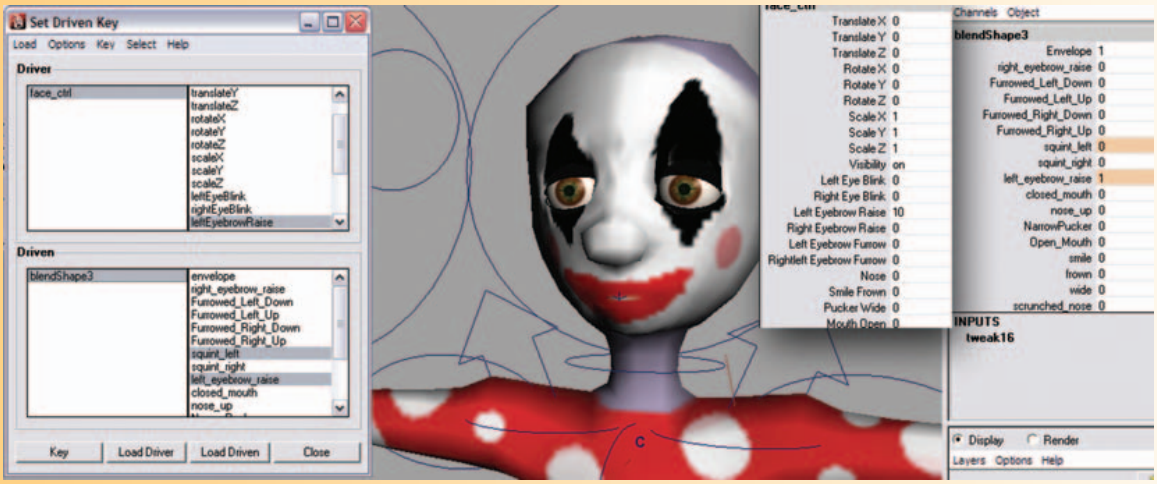
iii. In the *Driver* section of the *Set Driven Key* window, **click** on *face_ctrl* to **select** it.

iv. In the channel box, change *leftEyebrowRaise* to "**10**".

v. In the *Driven* section of the *Set Driven Key* window, **click** on *blendShape1* to **select** it.

vi. In the channel box, change *left_eyebrow_raise* to "**1**".

vii. In the *Set Driven Key* window, **click** "**Key**".



Setting the second key so that when the *leftEyebrowRaise* is set to "1", the blend shapes cause the left eyebrow to raise.

viii. In the *Driver* section of the *Set Driven Key* window, **click** on *face_ctrl* to **select** it.

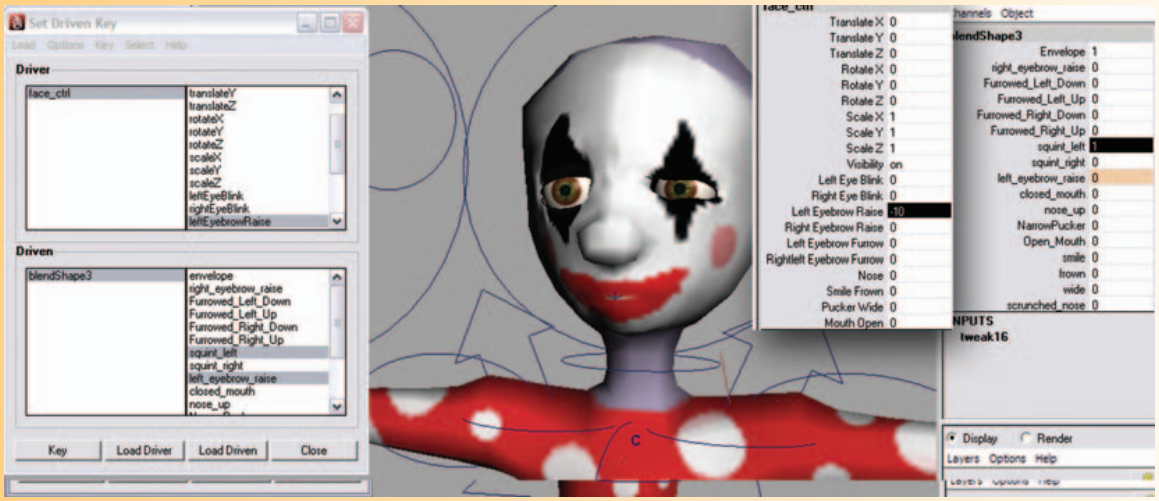
ix. In the channel box, change *leftEyebrowRaise* to "−10".

x. In the *Driven* section of the *Set Driven Key* window, **click** on *blendShape1* to **select** it.

xi. In the channel box, change *left_ squint* to "**1**".

xii. In the *Set Driven Key* window, **click** "**Key**".

xiii. Repeat these steps for the *rightEyebrowRaise*.



Setting the third key so that when the *leftEyebrowRaise* is set to "−1", the blend shapes cause the eyes to squint.

17. Make the remaining attributes function using *Set Driven Key* the same way using the following as Driver and Driven:
    a. Driver: *face_ctrl :* "leftEyebrowFurrow".
    b. Driven: *blendShape1*: "left_furrow_up" and "left_furrow_down".
    c. Driver: *face_ctrl*: "rightEyebrowFurrow".
    d. Driven: *blendShape1*: "right_furrow_up" and "right_furrow_down".
    e. Driver: *face_ctrl*: "nose".
    f. Driven: *blendShape1*: "scrunched_nose" and "nose_up (optional)".
    g. Driver: *face_ctrl*: "smileFrown".
    h. Driven: *blendShape1*: "smile" and "sad_frown".
    i. Driver: *face_ctrl*: "puckerWide".
    j. Driven: *blendShape1*: "narrow_pucker" and "wide".
    k. Driver: *face_ctrl*: "mouthOpen".
    l. Driven: *blendShape1*: "open".
18. Save your scene file. Name your scene *06_asgn07.ma*.

Assignment 6.8: Creating a Control System for the Tails, Ears, and Other Things That Move (Neckties)

Well congratulations, you have successfully learned many of the tools and techniques used to create a control rig for a 3D character. Depending on your design, however, you may still have a few body parts that need to be addressed. So here's a chance to apply what you've learned.

If your character has ears, it is usually a good idea to add some type of joint system to control them. Ears can be a very expressive part of a character's body and really helped an animator show great emotion from a character. Usually, three or four joints may be enough to create controls for short ears, but if the character has lengthy ones you may want to include a few more. Remember to check the local rotation axis and reorient the joints if necessary. You can use Set Driven Key or an IK chain with a controller to control the ear joints. Just make sure, if you're using IK, to parent the ear controller to the existing head control. This way the ears won't stick behind the character as he moves around.

Tails, antenna, and neckties can be easily controlled using the same system that we created for the spine. Usually the Spline IK creates a very fluid movement for long joint chains. Again, make sure the controllers for these body parts are parented to the existing control system. Tail controllers should be parented to the upper body control, antenna would most appropriately be parented to the head control, and a necktie of course would probably work best on the net control.

Whatever you are rigging, applying the tools that you've learned in this chapter can provide you with much of the control that will be needed when animating. This chapter has also provided you with a great foundation for learning more advanced techniques.