Using Armatures to Deform Meshes

Blender’s animation capabilities are great for most object animation except when you want to animate something bending like a person in motion or a tree bending in the breeze. This calls for a mesh to deform which can’t be done with traditional modifiers. We can deform a mesh in 2 ways in Blender. One way is to create a skeleton and have it deform a mesh (armatures) and the other method is to move the mesh vertices in edit mode and create sliders that deform the mesh (vertex keys). This chapter deals with creating armatures. The armature feature in Blender is constantly under development. For this discussion, I will stick with the fundamentals. More information can be found at www.blender.org or at www.blenderartists.org.

The first thing you need to do is create a mesh that has a few groups of vertices where you would like the object to bend. Any mesh will work and to get additional vertices you can either **extrude** or **subdivide**. Be careful not to create too many vertices. It may slow your model down considerably. Let’s use a cylinder to create an arm. I will use a cylinder set at the default divisions of 32. Next, I will change to a front, ortho view and box select the top set of vertices and **Extrude** them up. I prefer to use extrude rather than subdivide to keep the vertex count down as low as possible. As I extrude the vertices, I am also using **Scale** to shape them.

Next, place the 3D cursor directly at the bottom of the shape you just made. Hit “**Shift-A**”, to add an “**Armature-Single Bone**”. You will immediately see a bone begin to form at the cursor location. Enter **Edit Mode** and type “**G**” to grab the top of the bone and lengthen it to a desired size. Move your cursor up to lengthen the bone and click where you would like the joint to be. To create another bone at the top of the first one, press “**E**” to extrude another bone from the first one. If you run out of room to drag the mouse up, just click wherever and hit “**G**” again to move the end. To always adjust bones, you must be in Edit mode. Also, make sure you have the end of the bone select and not the entire bone. When finished, press **Tab** to exit edit mode. Double check the armature to make sure that the ends and joint are well aligned. To add more bones, enter edit mode again to extrude with the end bone selected.

Your next step is to create a **Child-Parent** relationship between the mesh and the armature with the mesh being the **Child** and the armature being the **Parent**. While holding the “**Shift**” key, select the mesh first, then the armature. Press “**Ctrl-P**” to make parent. Select the option “**Armature Deform**” and “**With Automatic Weights**” so the computer will figure out which vertices to deform to which bones. If it's not right, we can fix this later.
To test the armature system you just created, select the armature only and switch from Object mode to Pose mode. Right-click on the top bone and rotate it. It should deform the mesh as you rotate the bone. By rotating the lower bone, you will rotate the entire mesh.

**RoboDude Says:**
If you ever need to return to edit mode for the mesh or the armature after posing, they will temporarily return to their unposed states.

**Creating Complex Armature Chains:**

Extruding bones as we did works well for simple chains, but if you want to make more complex chains and skeletons, you'll need to know a few more things than just extruding from the end of a chain. You will notice that the 1st bone you created is the master parent for the system and you can also extrude from the bottom of that bone. The problem is that, in Pose mode, this bone will not be automatically parented to the master bone. You can also make entirely different armature chains, then Join them together using “Ctrl-J”. In pose mode, these will also not function with the entire system. So how do you correct these child-parent issues?

First, you need to know the Name of each bone. In Pose mode, you can RMB click on a bone and the name will display, but you can have all the bone names display on the screen if you turning on an option in the Object Data panel (now displayed as an armature). The example shown consists of 2 different armature objects that have been joined together in Object mode using “Ctrl-J”. When you enter Pose mode, they do not move together. In the Display panel, you will see an option to display the names on the screen. You will also see some options to change the way the bones look. To correct the parenting issue, go to the Bone panel, enter Edit mode and find the option for Parent. Select the bone you wish to parent to.

RoboDude Says:
If you ever need to return to edit mode for the mesh or the armature after posing, they will temporarily return to their unposed states.
Animation Tips:
When animating an armature, there are a few techniques that can make your life easier and were discussed previously in the animation chapter, but will be reviewed here. Since you have so many bones to animate, it makes sense to use the Automatic key-frame feature. It is also helpful to use the Rotation transform manipulator and the time line to advance through time. It may be helpful to review the chapter on animation for assistance. Below are some of the basic setting that we use for animating armatures:

Screen Layout: Switch to animation
Rotation Widget: Makes rotation easier

Time Line: Can move to key-frames quickly with controls
Transform Widgets: Turned on, set to Rotation and Normal alignment. Used to rotate bones by grabbing the right axis.
Automatic Key-frames: Record button on and set for Available. Remember to turn it off when you don’t need it!

RoboDude Asks: Why doesn’t my armature animation start correctly on frame 1?
When working with automatic key-framing, don’t forget to move every bone a little at frame one to set the initial animation keys, then move up through time to make your next move. A lot of people forget to add keys to frame 1.
Creating Bone Vertex Groups

Using the Automatic Weights option works great for simple meshes and armatures, but when bones and mesh vertices are close together or far from the bone, Blender will have a difficult time deciding how to join them. In the example to the right, you can see that some vertices for one finger have been grouped with the bone beside it. We can alter these vertex groups.

With the bone Names turned on so you can see which ones you need to effect, select the mesh and enter Edit mode. Editing bone vertex groups is similar to making normal vertex groups except that Blender already named a vertex group to match every bone for you when you made the child-parent relationship. If you go to the Object Data panel (mesh selected in edit mode), you will see the list of groups in the Vertex Groups panel. You will also see the standard “Assign” and “Remove” buttons below the groups.

To fix the problem, you will first need to select the vertices you wish to change, then select the vertex group that they are wrongly assigned to. Hit the “Remove” button to remove them from that vertex group. In this case, they are wrongly assigned to “Bone.004”. Now select the vertex group they need to be assigned to (Bone.010) and hit the “Assign” button. Exit edit mode, select the armature and move the bones in pose mode to test your groups.

For vertices that are close to a joint, they can be shared between multiple groups. You can also use the “Weight” slider to adjust the bone influence for other effects, but normally, it is set to 1.000.

Need to take your posed armature back to its starting pose? Change it from “Pose Position” to “Rest Position” in the Object Data.
Using Inverse Kinematics (IK) and Constraints

Inverse Kinematics is used when you wish to manipulate a skeleton by simply grabbing (G key) the end bone of a chain and moving it with all of the connected bones following along. Inverse Kinematics is constantly under development and has a lot of options available depending on what you want them to do. Visit www.blender.org for more details and review the wiki documentation.

To use Inverse Kinematics, select the bone at the end of a chain, go to Bone Constraints and add an Inverse Kinematics constraint. You can control the number of bones in the chain (Chain Length) and even give it a Target to point to like an empty. There are a few other options available including Influence (amount of effect). Once applied, you can use the “G” key to move the bone around while all those in the chain below it move in relationship to it.

Using Rotation Constraints:
When animating an object (like a finger), you may want to limit the direction and angle the object can bend or some “unnatural” things may occur when you try to add animation keys. Using Limit Rotation constraints can help with that. For the example to the right, I've applied a rotation constraint to the top 2 bones of the finger limiting the Y and Z axis to a limit of 0 degrees while the X axis can rotate from 0 to -60 degrees. This works because I switched the “Convert” from “World Space” to “Local Space”.

Copy Rotation:
Copy Rotation can make animating chains easier. In the example, I've created a single bone armature (not a bone in the existing armature) and child-parented it to the hand armature. This bone can be placed anywhere you wish. I have mine above the finger that it will control. Set it's rotation limits with the Rotation constraint discussed above, then add a “Copy Rotation” constraint to each bone that will copy it's movement. Select the control bone for the Target. In my case, I needed to Invert the motion to work correctly.

RoboDude Says: Press “Ctrl-A” to reset an object's rotation and scale. Works with individual bones in Pose mode. This helps a lot with these constraints!
In this exercise, you will be creating a simple hand using meshes and armatures. Start a new Blender file and call it “Hand”. As discussed at the beginning of this chapter, make a simple finger using a cylinder or filled circle. Extrude it to have 2 joints and looks something like this:

Now duplicate (Shift-D) the finger 4 times, scaling and rotating them into the shape of a hand. Don’t worry if it doesn’t look perfect - this is just a simple exercise. Try for something like the image to the left.

In order to add the palm of the hand, I used a UV Sphere and scaled it in the “Y” direction to make it narrow and used Proportional Editing to shape it to fit the fingers. Try shaping yours into something like this.

After shaping, select all of the meshes and Join them together (Ctrl-J).

It’s now time to add the Armature. We will only be animating the fingers opening and closing for this exercise so we only need bones in the fingers. Place your 3D Cursor at the base of one finger. As discussed in the chapter, add an Armature, then enter Edit Mode and pull the top end of the bone to align with a joint. Extrude (E) 2 more bones to fill the finger. Exit Edit Mode and return to Object Mode.

Now Duplicate the armature (Shift-D) and place it in the next finger. Enter Edit Mode and move the bone joints to match the mesh finger joints. Exit Edit Mode and continue duplicating and adjusting armatures until all fingers are finished. When finished, use Ctrl-J to joint all the armatures together.
It's now time to create the child-parent relationship for the mesh to the armature. Use "Armature Deform" and "Automatic Weights" as discussed in the chapter when using "Ctrl-P" to make the relationship. Remember to select the Mesh first. It will also be helpful to turn on "Names" and "X-Ray" in the Object Data panel. Feel free to change the display type in that panel as well if you wish.

Enter Pose Mode and test your armature out to see if it works well or if you need to adjust vertex groups. If it needs adjustment, follow the steps of re-assigning verticies to the various bone groups discussed in the chapter.

When everything looks good, create a simple 200 frame movie of the hand moving.

**Challenge Exercise:**

Add constraints to limit rotation or control armatures with rotation copy constraints.

**Call the instructor when finished**