

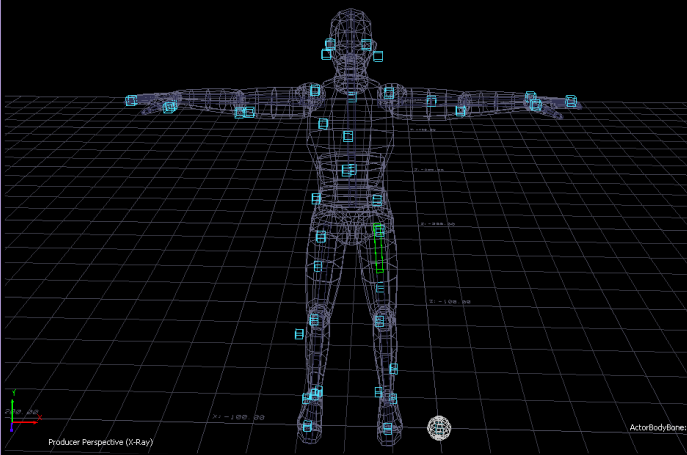
1

Motion Capture



Motion capture is a process by which the positions of small reflectors placed on an actor's or object's body are captured by special cameras. This allows one to quickly capture extremely intricate performances in a fraction of the time required by traditional animation. This process is a popular technique in large effects houses, and is close to becoming available to a wider market.





Motion capture is a tool, while traditional animation is a process. There will always be a place for traditional animation. The myth that motion capture will replace traditional animation is just that, a myth. In reality, more often than not, the motion capture data, after it is captured, is sent to the animator and is often augmented or tweaked in some fashion or form. Although the data can be used straight from a motion capture session, an animator will usually alter it depending on the shot or sequence that it is to be used in.



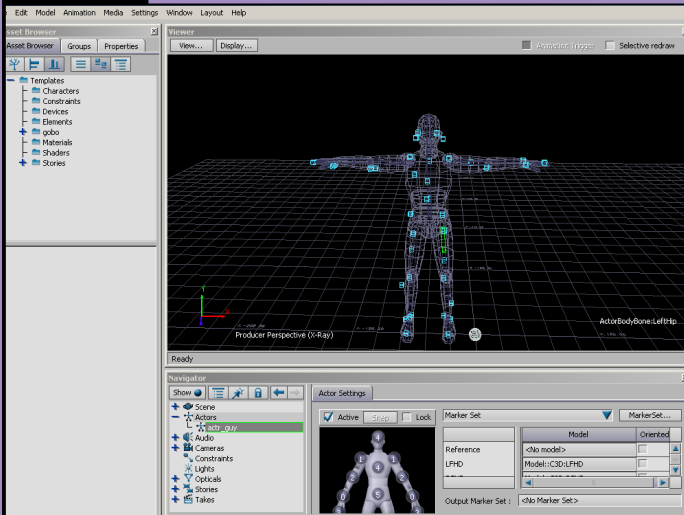


There are instances when you may need to traditionally animate an object or un-human like creatures. Other characters that are not well suited for motion capture are cartoony characters. Usually any character that needs exaggerated motion, i.e. squash and stretching, is animated traditionally. What motion capture is best suited for though are any lifelike, realistic background characters, crowds, even stunt performers or in that unforeseen event when an actor or actress is no longer available to film a scene, and a digital double is called for.



4

Editing Motion Capture



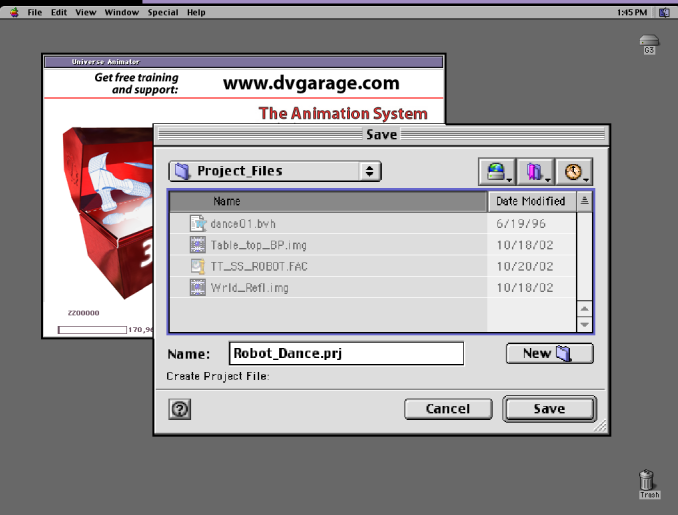
Editing motion capture in Electric Image is difficult. If you're really serious about it, Kaydara's Motionbuilder™ or Credo Interactive's Life Form™ applications are really the best place to work with the data. Of these, Motionbuilder is the stronger of the two. If you have a data set that works to your satisfaction though, it can then be implemented in EI very easily.

In this tutorial we will walk you through adding existing motion capture data to a prebuilt model. At the end of the tutorial we will mention some key information on how to link up your own motion capture data to your own model.



Overview





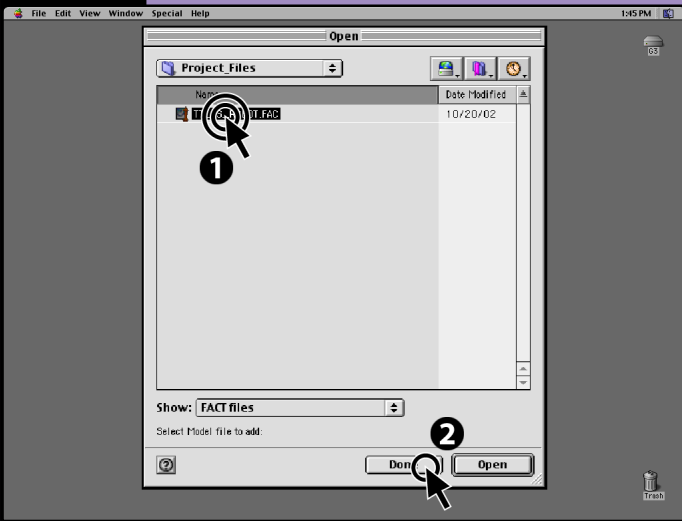
Launch Electric Image Animator.

Press [**CMD/CTRL+N**] to start a new project file.

When Electric Image prompts you to name and save this new project, name it "Robot_Dance.prj" file, then navigate to the Motion_Capture Tutorial folder and save it in there.

Note: Macintosh keyboard commands are indicated in **red**. Windows keyboard commands are indicated in **blue**. Some files may need to be manually located while loading.





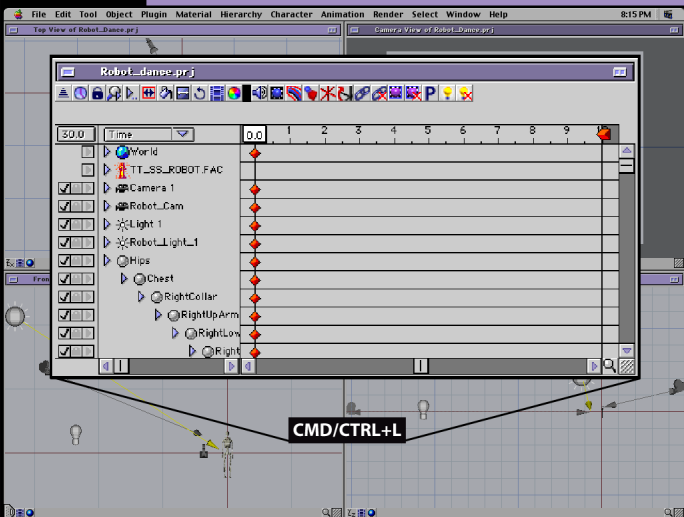
After saving, you will be prompted to load your FACT files into this new project.

Locate the TT_SS_ROBOT.FAC file and **[CLK]** Open.

If there were any more models to add, you could continue adding them into the project, but for this exercise, there are no more to add, so **[CLK]** the Done button

Note: you can save cameras, lights, geometry and links to texture maps in a model file. However, you cannot save motion data (animation).





Animator then generates the scene loosely based on the size of the FACT file we loaded. Before we proceed, we need to remove two unwanted elements: a Camera and a Light.

Open the Project window [**CMD/CTRL+L**].

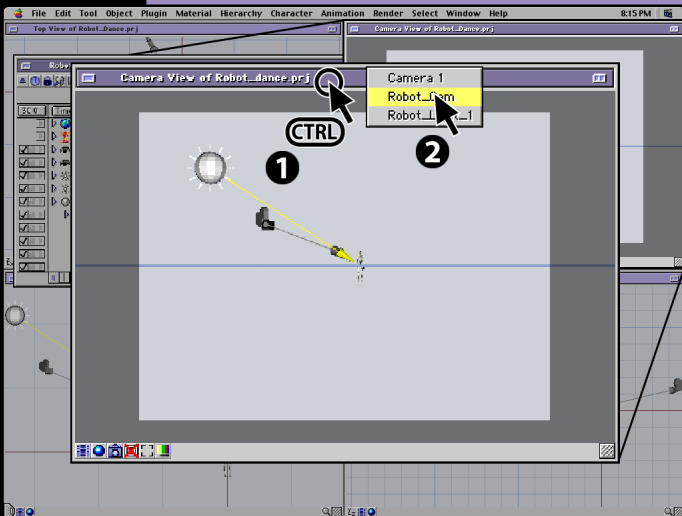
Animator defaults to adding a Light and a Camera into every new Project file. Since our FACT file contained a prebuilt Camera and Light, we do not need the default Camera and Light in our scene.

However, before we delete the Camera, we need to switch our viewing angle from the default Camera to the Robot_Cam...



8

Switching the Camera View



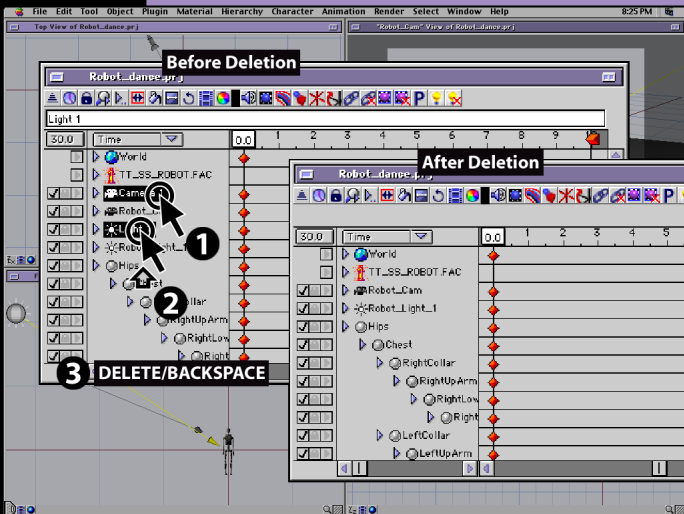
In the Camera View window, [**CTRL/R+CLK**] on the Camera View window header bar.

In the pop-up menu, select the Robot_Cam Camera.

The Camera View window should now be looking at the set from a higher elevation.

We can now delete the default Camera and Light....



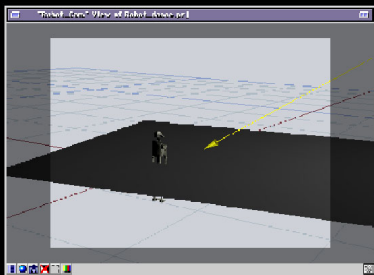


So, in the Project window, **[CLK]** on Camera 1, **[SHFT+CLK]** on Light 1, then press the **[DELETE/BACKSPACE]** button.

You should only see the following in the Project window: World, TT_SS_ROBOT.FAC, Robot_Cam, Robot_Light_1, and the model objects of the Robot itself.

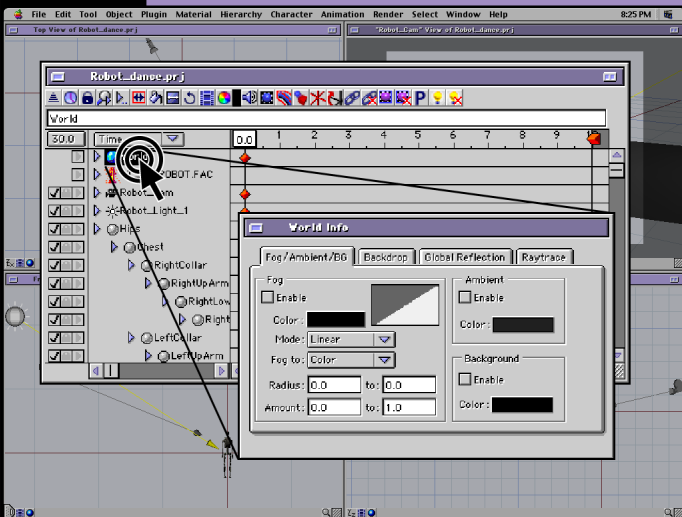
Now on to the exercise....





The majority of this scene is already set up for you. What we will be doing is adding the finishing touches to it. Our goal will be to comp the robot into a background plate, adding some motion capture data to the robot and setting up a shadow pass to really help sell the illusion that the robot is dancing on a table.



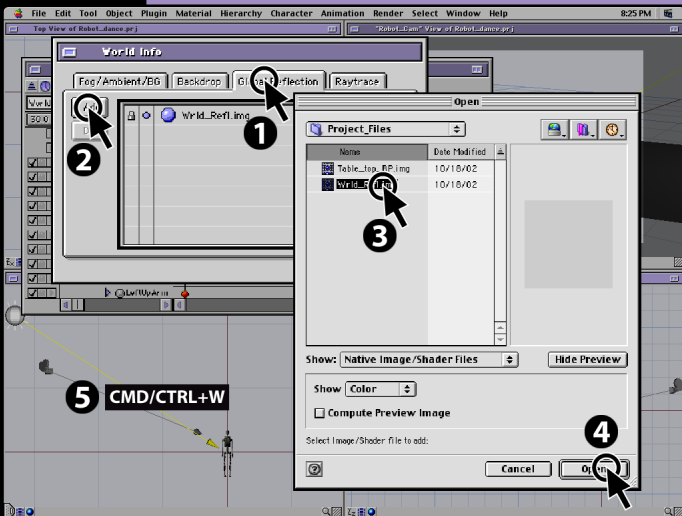


Still in the Project window, **[DBL+CLK]** on the "World" layer.

The World Info dialog box opens up.

Note: This dialog box is where you can set up various attributes that help define your scene. You can set up global fog to reproduce distant haze (atmospheric effects) which, when used effectively, can add depth cues to your environment. You can also set the backdrop of your scene, unless you are compositing your 3D objects into one later on. The one tab we are going to pay attention to right now is the Global Reflection tab.





[CLK] the “Global Reflection” tab.

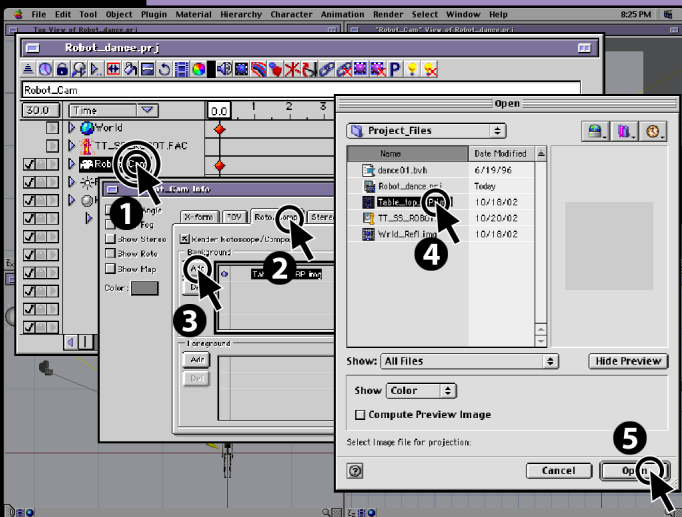
[CLK] on the “Add” button.

In the Open dialog box, navigate to the Motion_Capture Project_Files folder and select the “Wrld_Refl.img”.

Press **[CMD/CTRL+W]** to close the World Info window.

Note: This reflection map is for the robot. It is a photo taken of the outdoor environment where we will be comping in our robot dancing on a table. The map will make the robot look like it’s actually on the table.





[DBL+CLK] on the "Robot_Cam".

In the Project window, [CLK] on the "Roto/Comp" tab.

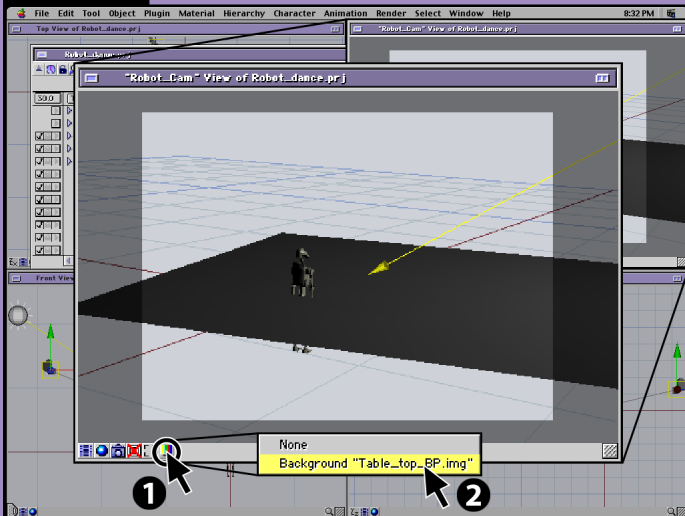
In the Background section, [CLK] on the "Add" button.

In the Open dialog box, locate and open the "Table_top_BP.jpg".

Close the Robot_Cam Info window when you are done.

Note: This is the background image that we will have the robot dancing in.



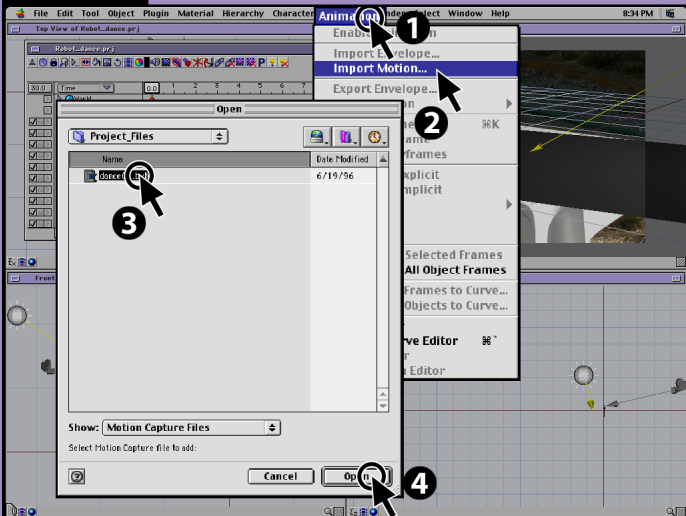


In the Camera View window, **[CLK]** on the color bars icon (the Rotoscope button) to select the roto/comp image that we loaded in the previous step.

Select Background "Table_top_BP.img".

Now we can see the background that our dancing robot will be in.





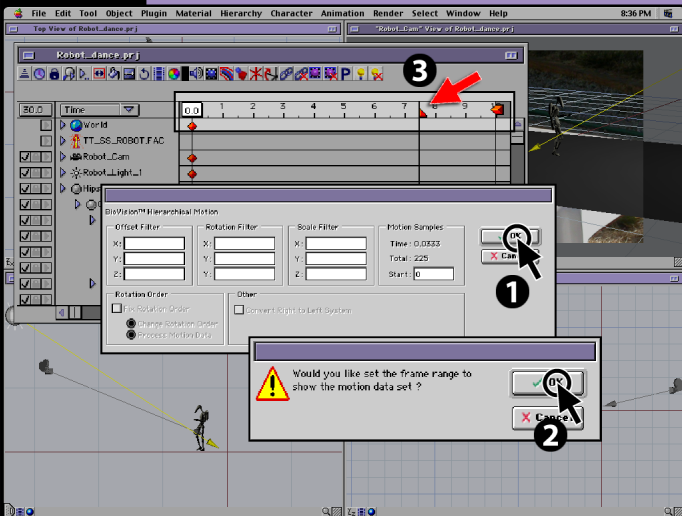
Now here is the fun part--adding the motion data.

In the main menu, select Animation > Import Motion.

In the Open dialog box, navigate to the Project_Files folder and select the "dance01.bvh" file.

[CLK] on the Open button or press [RTRN].





In the Biovision dialog box, **[CLK]** OK or press **[RTRN]**. Do not be tempted to enter anything in these fields. If you do, you will offset the motion data in the file which will cause headaches later on. There may be times when you need to do this for a model in your scene. If you do, proceed with caution. It is always best to add motion data to a neutral posed model centered at the origin.

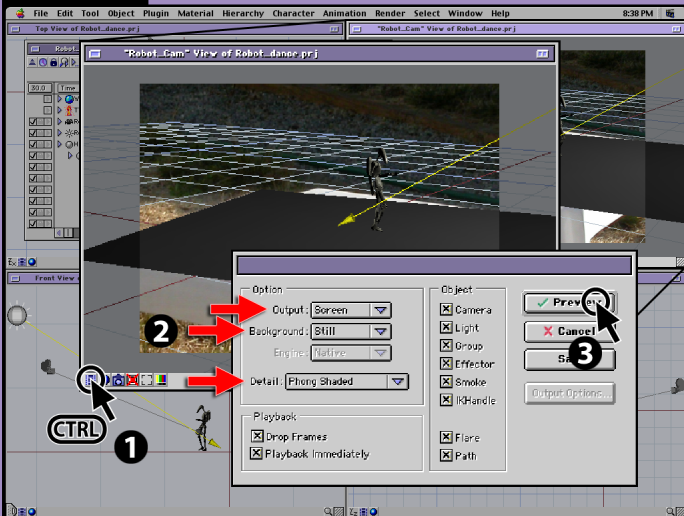
In the next warning dialog box, **[CLK]** OK or press **[RTRN]**.

This applies the Motion Capture to our robot. If you notice in the Project window, a red half triangle appeared atop the timeline. This is the result of OK'ing the last dialog box and it marks the end of the motion capture animation data.



17

Preview the Motion



In the Camera View window, **[CTRL/R+CLK]** on the Preview button (button on the far left).

Set the Output option to Screen, background to Still, and Detail to Phong Shaded.

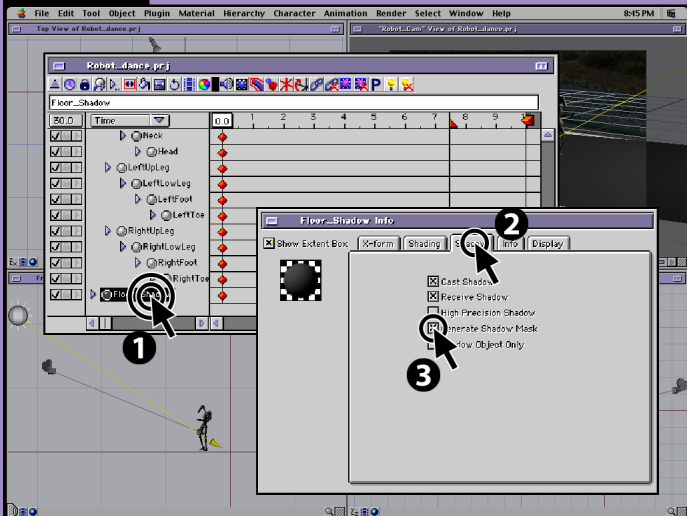
[CLK] on the Preview button.

Note: While everything seems to be in order, we have one last thing to deal with - adding a shadow...



Applying the Motion Data





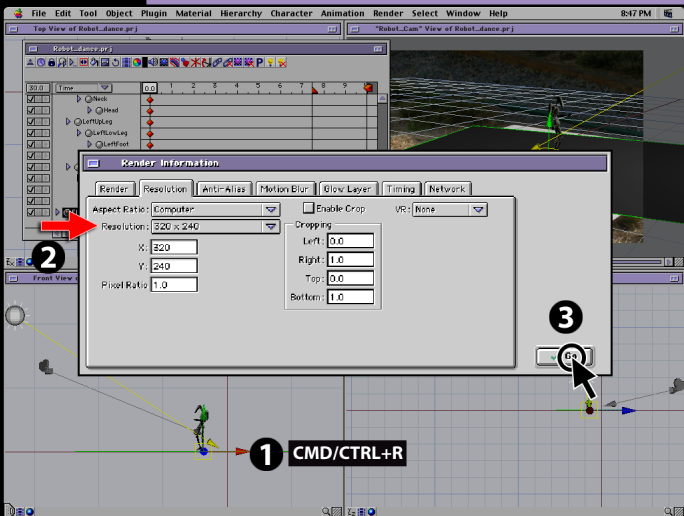
In the Project window, scroll down to the bottom of the list and **[DBL+CLK]** the "Floor_Shadow" object layer.

In the Shadow tab, turn on "Generate Shadow Mask".

Close the Floor_Shadow Info window when you are done.

Note: This will cause the plane to render only where shadows hit the surface. This fake shadow makes your robot look like it's interacting with the environment. While there are other methods of creating shadows, to control the shadow when compositing directly within EI, it's easier to create the shadow in this manner.





Press [**CMD/CTRL+R**] to open the Render window.

Set the Resolution to 320x240.

[**CLK**] "Go" to begin rendering.

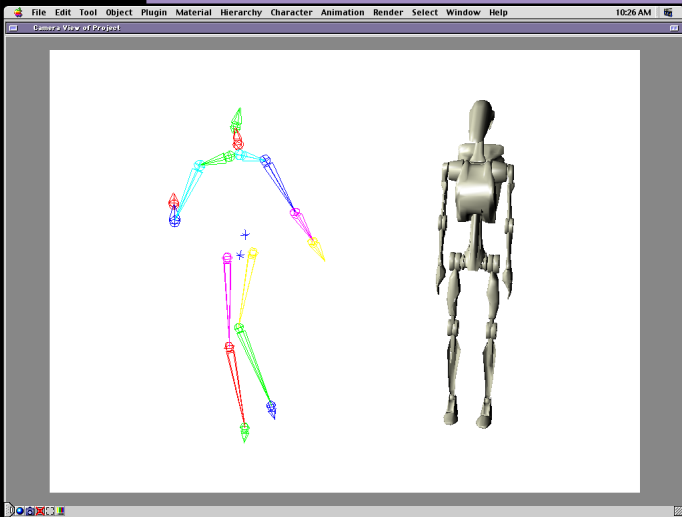
And that is all there is to this...

Note: You will need to save your project file and direct and name your render. Later, you can open it in After Effects and Quicktime (with the proper plug-ins). You can also set Render Output to Quicktime if you don't have the other options.



View the Animation



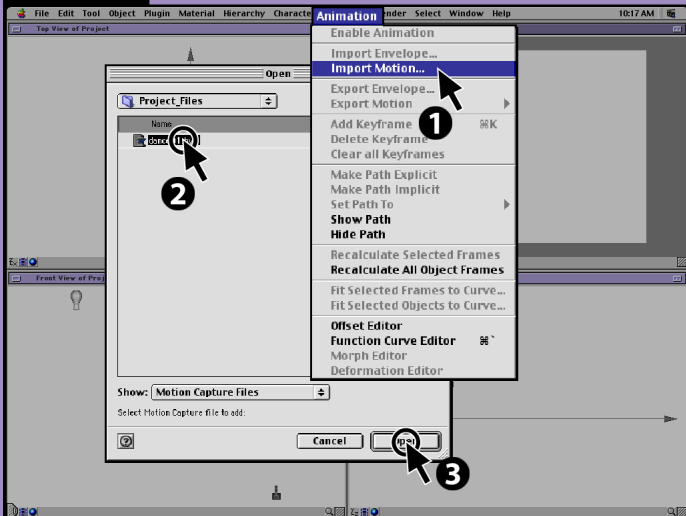


While this tutorial went through adding existing motion capture data to an existing model... suppose you would like to add your own motion capture data to models that you have created?

How do you set the model up to accept the motion capture data?

The key is naming the objects of the model and mapping the motion data to those objects. Let's say you already have the motion capture data file, how do you know what to name your objects? Let us explore that...





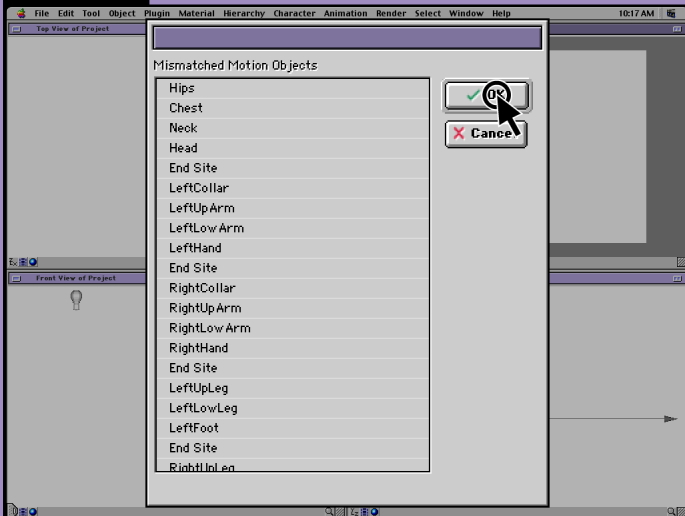
Since we already have a motion data file (.bvh) and we want to apply this to a model that is already built, we somehow need to know the naming convention the .bvh file uses.

Close and save the current project, then create a new blank EI project file (don't add or import any models).

Select Animation > Import Motion.

Locate your file. If you don't have your own file, use the "dance01.bvh" file.

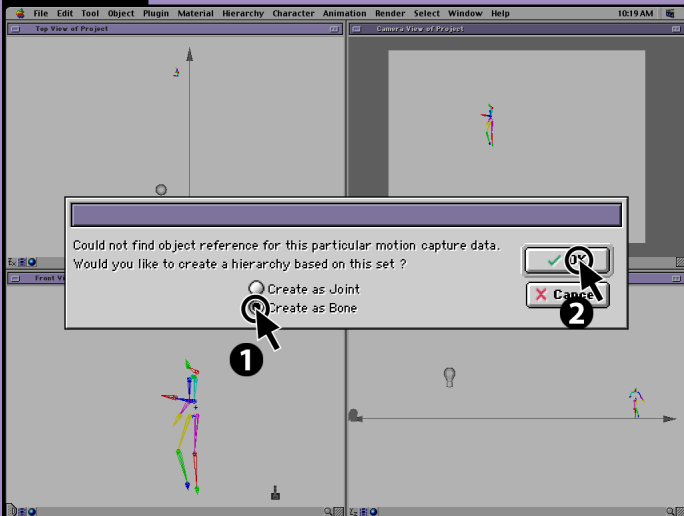




The dialog box that appears next is letting you know that there is no model in the project file that it can attach the motion data to. The list that you see is the naming list of the objects in the motion data file. As you can see, the top name is called "Hips". The BVH file format only contains translational hierarchical information. Unfortunately this list does not show that.

[CLK] OK and move to the next dialog box.





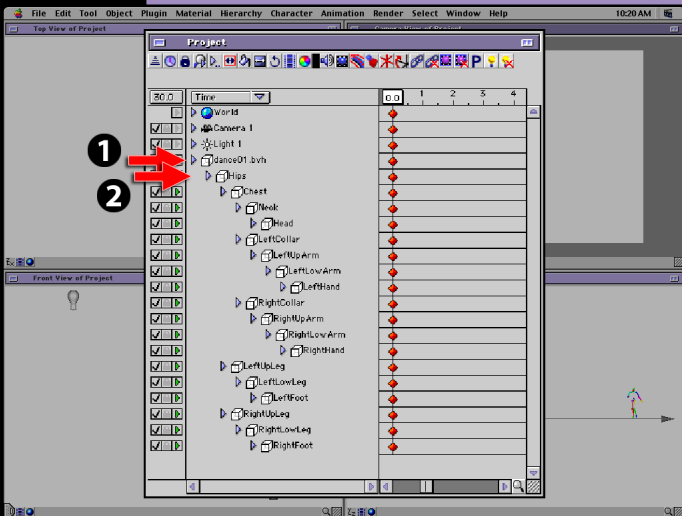
The next dialog box is asking to confirm that no object exists to attach the data to and if you still want to open this file. Since there is no model, EI would like to know how you would like to create a hierarchy based upon the data.

Select the Create as Bone option.

[CLK] OK.

[CLK] OK on the next two dialog boxes that come up.





You should now see a boned figure posing in the empty scene.

Press **[CMD/CTRL+L]** to bring up the Project window.

Resize the window to clearly see all of the objects created.

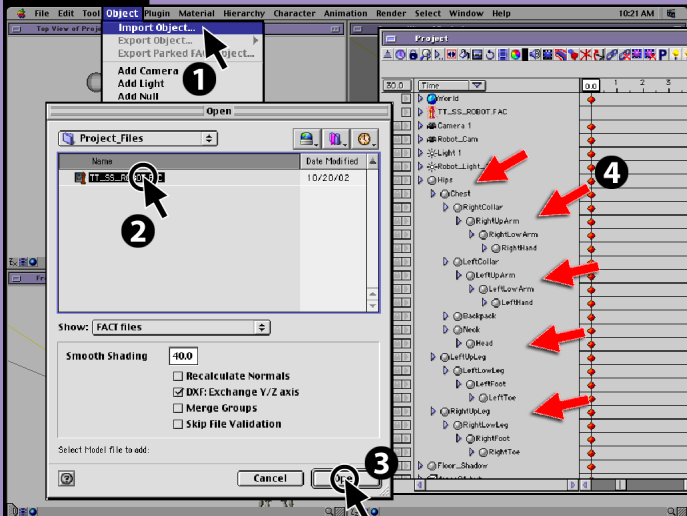
This is the linked hierarchy of the motion data file, complete with the expected names of the objects it wants to match to.

The first link is just the name of the file. The key link though is the one directly attached to this link: "Hips". This is the main link that holds the data together. BVH translational data always starts from the pelvic area first, then branches off from there.



Wrap Up





As you go through the list, you can see the naming conventions used and what is linked where.

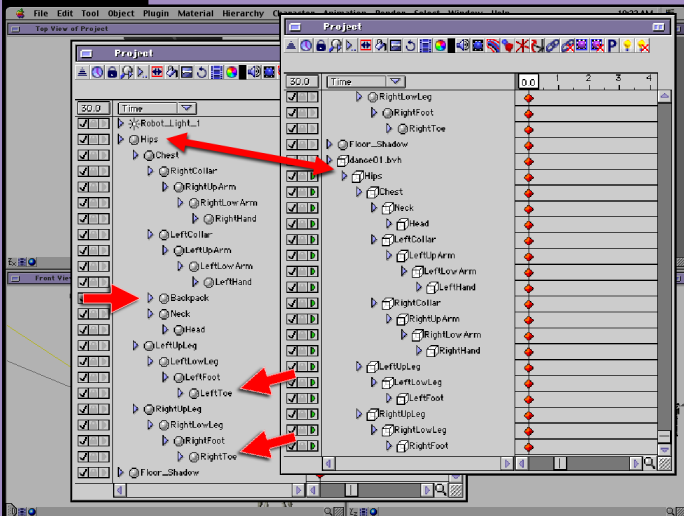
Select Object > Import Object.

Import the TT_SS_ROBOT.FAC model.

[CLK] the Done button.

Reposition the Project window to see the Robot's hierarchy. Look at the naming convention. It should look very familiar, like the motion data file that we just imported.





As you do a one to one comparison you will see some additional objects on the Robot that are not on the motion data file. This is perfectly fine because of where they fall in the parent/child link structure. Those objects are being driven by a parent object that is correctly named on the motion data file.

So... armed with this information, you should now be able to play around with motion data files and link your models up appropriately. If you have any questions or concerns, please visit us on our forums, where the discussion of character animation has begun.



Wrap Up

