

▼ How does it Work?

▼ Tutorial: Vampire Alive!

▼ Other Techniques

▼ Project Constraints

▼ General Concepts

Why Pivot Points are Important

Why Hierarchy is Important



## Creating Animation for LightWave 3D in Life Forms Studio 3

Introducing a new way to animate characters for LightWave 3D.

Character Animation is one of the most rewarding albeit challenging type of animation you can undertake as a 3D artist. Now, there is a new way for LightWave users to animate. You'll want to read this document if you are looking for ways to:

- Create more motion intensive projects.
- Apply mocap data to models in LightWave scene files.

This tutorial describes how we set up the project "Vampire-alive!". Vampire Alive was created with a model created in LightWave. Using bones, we applied mocap sequences from Life Forms Studio 3.0 to the mesh and rendered in LightWave.

You will need **Life Forms Studio 3.0** and LightWave **5.6**, and of course the **new LightWave plug-ins** from Credo Interactive Inc.

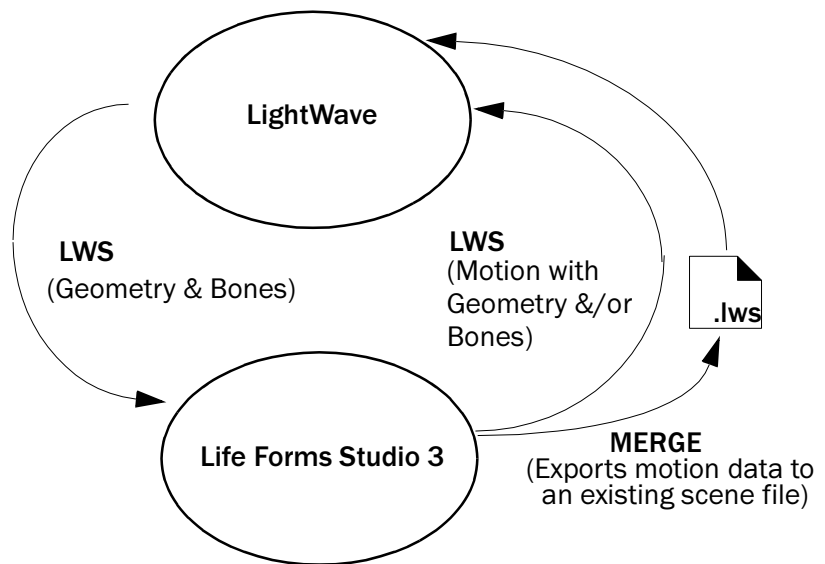
## How does it work?

---

Life Forms imports and exports the LightWave's scene file (.lws). This gives you maximum freedom in your 3D projects. You will be able to:

- Insert motion into existing scene files using the Merge option.
- Use Models and Motions in Life Forms PowerMoves Library for LightWave projects.
- Use LightWave's ability to deform meshes by using bones.

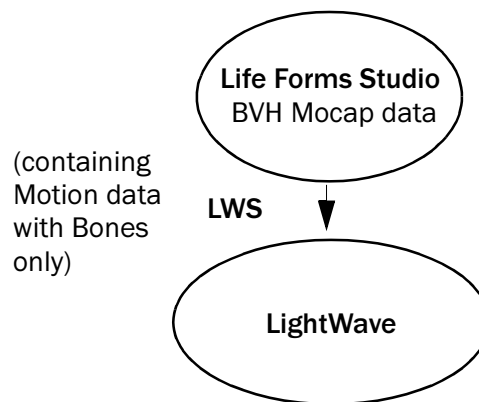
The diagram below shows how you can move data between Life Forms and LightWave. As you can see, there are many ways to efficiently create a motion-rich scene file. You can begin your project in LightWave by building custom models. Or, you can start in Life Forms by using the models in the Model Gallery and PowerMoves Libraries. You can choose between using bones or segmented models.



## Objective

There is no one prescribed way of using Life Forms to enhance your scene files.

This tutorial shows you how to animate bones for LightWave. The motion of the bones will deform and animate a mesh Vampire in LightWave. The illustration below summarizes how data moves between LightWave and Life Forms in this tutorial. For other ways of using Life Forms see **Other Techniques**.



## Boning a mesh

Critical to this tutorial, is knowing how to *bone* a mesh. Setting up bones properly to animate a mesh requires you to have a solid understanding of how Lightwave handles bones (and some patience too). The LightWave User Guide has excellent documentation on how to set bone strength, rest length, and on how to use joint compensation to control the way bones affect the mesh.

## Summary

In this tutorial, show you how to bring BVH motion data as bones into LightWave and apply it to a mesh model.

### **Important. Please note the following:**

- 1 If you have Life Forms Classic 3.0 or Life Forms 2.1 you will need to upgrade to the Studio version to use Life Forms with LightWave.
- 2 You will need latest Lightwave plug-in. Please contact us at [info@credo-interactive.com](mailto:info@credo-interactive.com).

## Conventions and Files used

This tutorial uses both LightWave 5.6 and Life Forms Studio 3. To make it clear which application to use at each step, the application is given on the left.

### Step 1

Life Forms

#### Export skeleton in the default position

In Life Forms, select a motion capture file. Clear all keyframes from the animation. Set the bones to have the same default position as the mesh — this is usually legs slightly apart and arms stretch out to the side. Then, export a single keyframe of the skeleton in its default position in a LWS file.

### Step 2

LightWave

#### Replace the Null root object

In LightWave, open the LWS file.

Then, replace the null root object of the skeleton with the Mesh object. You can do this in the Object dialog box by clicking the Replace Object button.

Don't worry if the Mesh is terribly distorted. This problem will be fixed when you set the rest position of the Arm bones.

### Step 3

LightWave

#### Resize the Mesh to fit the bones

Open the Modeler, and get the Mesh. You will now have to size and position the mesh in the modeler. Check each change using the Put button and switch to Layout mode. Make all bones Active to see how the bones interact with the mesh.

Iteratively size and position the mesh in the Modeler and viewing the effect in layout mode until the mesh and bones fit together.

Now you want to save the adjusted Mesh as a new LWO file. Name it so that you know which skeleton to use it with.

**Note** You may have to zero the rest position of the root bone, or conversely offset the mesh from it's zero position.

### Step 4

Life Forms

#### Adjust the Bone rest position

This is done by rotating the bones until they fit properly within the mesh. Note down the H, P and B values of the rotation. Open the Object dialog box, and click Object Skeleton. Select the joint then click Rest Position. Enter the H, P, B values and click OK.

Note that the bones no longer distort the mesh.

## Step 5

LightWave

### Export Motion data from Life Forms

Open the motion capture file from which you exported the bones. Export this file as a LWS file. In the Export dialog box, Click Bones only. Click OK to export.

## Step 6

LightWave

### Set the correct Rest position

In LightWave, open the Object panel. Click Object Skeleton, select the joint and click Rest Position. Enter the H, P, B values you recorded in step 4.

## Step 7

LightWave

### Apply adjusted Mesh to the motion

In LightWave, open the LWS scene file with the animated skeleton. Replace the null object with the adjusted mesh, and voila - you have an animated mesh.

## Step 8

LightWave

### Adjust the Bones strength

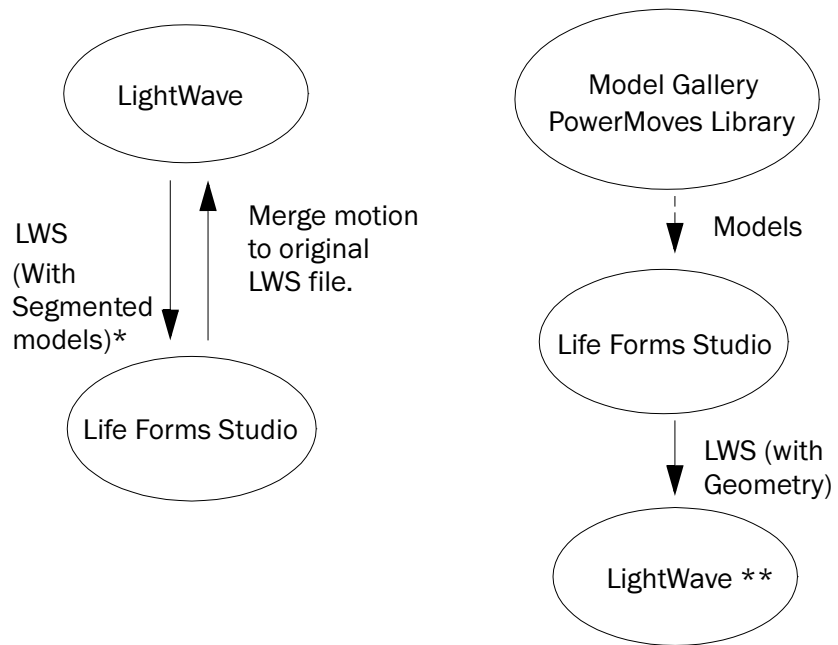
Refine the interaction between the bones and mesh by adjusting bones strength and rest lengths. Pinching can be controlled using Joint compensation.

## Other Techniques

---

### Using Segmented Models

Life Forms reads and writes geometry in the LWS file format, you can bring segmented models into Life Forms for animation. You can also start you project in Life Forms using models from the Model gallery and PowerMoves Library. When you export LWS files click Geometry. This will export the motion and geometry for these models. The following illustration shows how two ways of working with segmented models.



\* It is recommended that you delete any keyframes before importing the LWS file to Life Forms.

\*\* Can use Puppet Master to add skinning between joints.

## File Formats Supported

---

### What Life Form's reads:

As Life Forms opens a LightWave scene file it interprets only the information needed to let you animate models in Life Forms. It reads the following data from scene files:

- **Geometry** – both Single -mesh & Segmented (.lwo files)
- **Bones** – any bones you have set up in LightWave

This will mean that you can animate any articulated model in Life Forms that you can create in LightWave or that you already have in an existing project.

Life Forms will also read any keyframed rotational motion.\*

## What Life Form's writes:

Life Forms can export animation you created - either by using mocap data, procedurally using the Walk Generator or by keyframing to LightWave. You have several options:

- **Merge the motion data to a scene file** — If you have a scene file that contains objects that were animated in Life Forms, you can merge the animation of those objects to this scene file. This option exports motion without the geometry.
- **Export Motion with Bones Only** — Exports a new scene file. Allows you to export bones with animation. The benefit of this is that you can attach the bones to a mesh in LightWave.
- **Export Motion with Geometry** — Exports a scene file, creating new object files (.lwo) in the content directory for all models in Life Forms. Note any bones in the Life Forms will also be exported as objects.

## What Life Forms doesn't read or write

What it doesn't read are cameras, lights, textures and colors. When it reads in the model it will ignore any inverse kinematics (IK) settings you've set up because Life Forms has its own IK system. Bearing these factors in mind you should:

- Use the Merge command to add motion data. The Merge command inserts just the motion data to the.lws. The other scene elements are unaffected.
- Organize the project so that you color and apply textures to models, set up cameras, lights and other scene elements after creating motion in Life Forms
- For existing projects, remember to turn off all IK settings and goal objects before saving the file for Life Forms.
- 

## Project Constraints

---

- 1 Models must be polygonal meshes
- 2 Models must have only one null object
- 3 Only the root of a hierarchy may contain translations
- 4 Importing Scene files with Motion data into Life Forms is NOT recommended.
- 5 Merging animation to Scene files that contain motion data is NOT recommended.

- 6 Life Forms does not read Lightwave's IK settings from the Scene file. It is recommended that you disable IK set up and remove goal objects before saving a Scene file for Life Forms.

## General Concepts

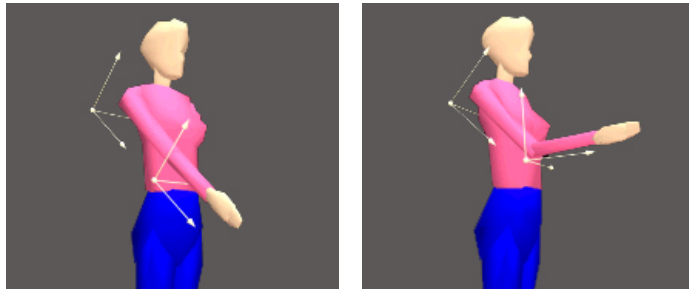
---

### Setting up Segmented Models for animation

For Life Forms to animate a segmented or articulated model, it must have correct pivot points and its joints must be linked in a hierarchy. In the Modeler mode in LightWave, objects can be parented to others to set up a hierarchy. You can also set the pivot point for each object. You should rotate the joints in layout mode to check that the pivot points are appropriate for the model.

### Why Hierarchy is Important

The axis of each joint is relative to the position of the joint. Another way of expressing this concept is that each joint has its own **local axis**. The axes of a joint changes with the position of its parent joint (e.g. the upper arm is the parent of the lower arm)..



Some motion data file formats store **relative** instead of absolute rotation angles, for example BioVision BVH format. Relative rotation angles are the angles of rotation given relative to the parent joint's local axes. When motion data is recorded in this way, the model that the data will be applied to must have the same hierarchy as the data in the motion file.

### Why Pivot Points are Important

To understand how pivot points affect the motion applied to the model, you have to understand how motion data is described. The position of a joint is described as the angle of rotation around the joint's pivot point. When motion data is applied it rotates the joint around the pivot point defined in ElectricImage. Thus, when the pivot points of the model are wrong, the motion data will rotate the



joint incorrectly. It may cause the joints to over or under rotate, or it may cause joints to break off from adjacent joints.

*Important*

This document is still being worked on. This document will be updated as more complete information is available. Your contributions and feedback are welcomed. Please send them to [info@credo-interactive.com](mailto:info@credo-interactive.com).