

# JWildfire tutorial on: the dancing-flame-module

by thargor6 (<http://thargor6.deviantart.com/>)

Version 0.1

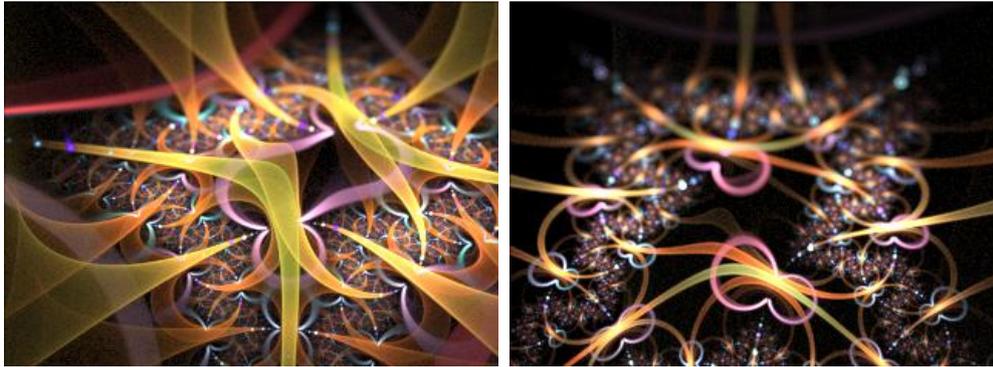
The following tutorial describes how to create sound-synchronized movies or short animations using the dancing-flame-module of JWilder.

All you will need is a recent copy of JWilder (Version 0.66 or higher is required) and some mp3 song. Please check the copyright of the song before uploading your movie later (together with the sound) to Youtube or some other video platform. Otherwise this may not only be a violation of a law, Youtube will probably replace the song, which will be not the greatest fun in case of a sound-synchronized animation ;-)

## Table of Contents

1 Introduction and basic concepts .....	3
1.1 The basic concepts .....	3
1.1.1 Motions .....	4
1.1.2 Flame properties .....	4
1.1.3 Motion links.....	5
1.1.4 Motion hierarchies .....	5
2 Basic documentation.....	6
2.1 Getting started with a dancing-flames-project .....	6
2.2 The flame-pool .....	6
2.3 Adding sound (mp3) .....	7
2.4 Adjusting the projects FPS (frames per second) .....	7
2.5 Previewing the show .....	7
2.5.1 Adjusting the speed of the realtime-preview.....	8
2.6 Switching between flames during the show .....	8
2.7 Recording a show .....	9
2.8 Rendering the flames .....	10
2.8.1 Using the batch-renderer of JWildfire.....	10
3 Animation examples.....	12
3.1 A simple rotation .....	12
3.2 Animation by using a manually created spline-motion-curve .....	14
3.3 Finally: sound-synchronized motion :-)	18
3.3.1 Motion damping (smoothing) .....	18
3.4 Using motion-hierarchies .....	19
Appendix: The flame used in all examples .....	20

# 1 Introduction and basic concepts



The dancing-flames-module is a very powerful and flexible program part of JWildfire to create animations of flame fractals.

What can you do:

- animate anything! (i. e., any property of a flame which can be accessed in the flame-editor can be animated)
- synchronize motion with sound (a frequency spectrum computed by a FFT can be used to link motion with "beat")
- create motion curves (amplitude over time) for certain properties by using a spline editor or apply a predefined motion (basing on a certain formula, like "Sawtooth")
- group motions together (e. g. have a spline-motion-curve which is slightly altered by beat")
- watch the motion in realtime and record it to be processed again to create frames at accurate framerate
- use any number of flames in your movie and switch between in realtime during the recording step (this is like a dj's job)
- load and save projects

## 1.1 The basic concepts

The basic concept of the animations in the dancing-flame-module is both very strong and very simple and contains of four building blocks:

- motions
- flame properties and
- motion links
- motion hierarchies

### 1.1.1 Motions

You can define any number of motions (which are not related to anything in the first step), e. g. :

- a rotation with frequency  $x$
- "beat" data of a certain channel from the FFT
- a manually created motion curve using the spline-editor



Each motion has special parameters which can be accessed through a property table. There are some parameters which can be found by all motion-types.

Motion	Start	End	GENERAL
SawToothMotion - 303705...			amplitude 0,07
FFTMotion - 1953613337	0		avgSize 2
SplineMotion - 1501875976			endFrame
SplineMotion - 506741794			fftChannel 3
SplineMotion - 2036671403			offset 0,073
FFTMotion - 1594420688			parent
			startFrame 0

### 1.1.2 Flame properties

Each flame has a tremendous number of properties. Most of them can be easily be animated, e. g.:

- camera angle
- zoom factor of transform 2
- Julia index of variation "julian" in the final transform

You have do not to know all of those, they can all be accessed through a tree-view in the graphical interface.

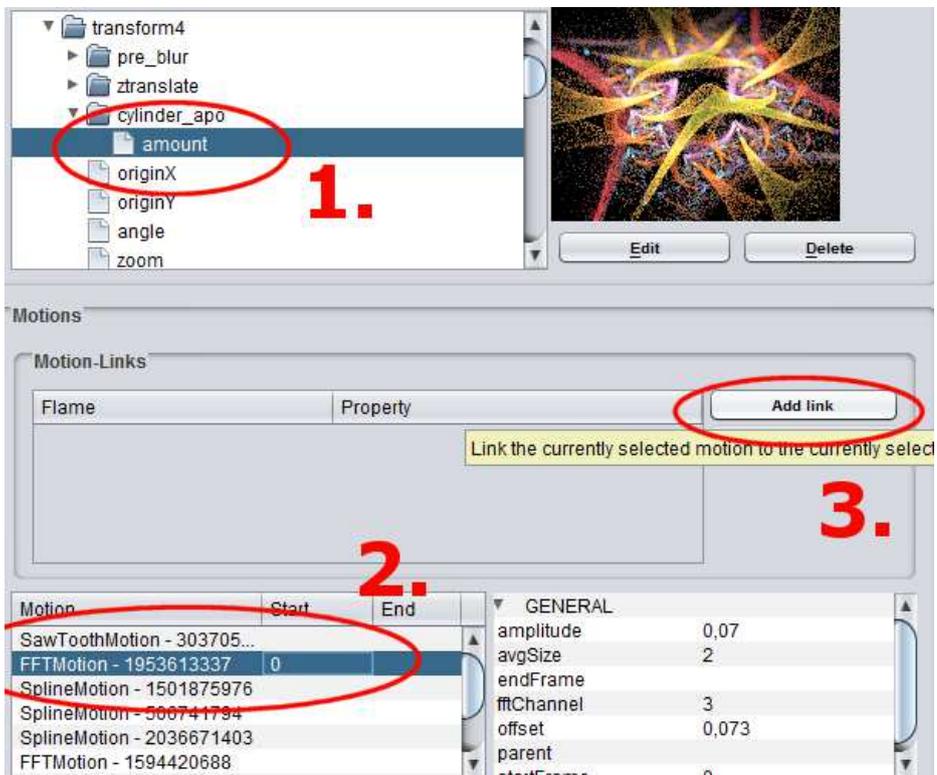


Each flame forms its own top-level-folder in this view. There a sub-folders for:

- each transform
- each final transform
- gradient parameters and
- shading parameters

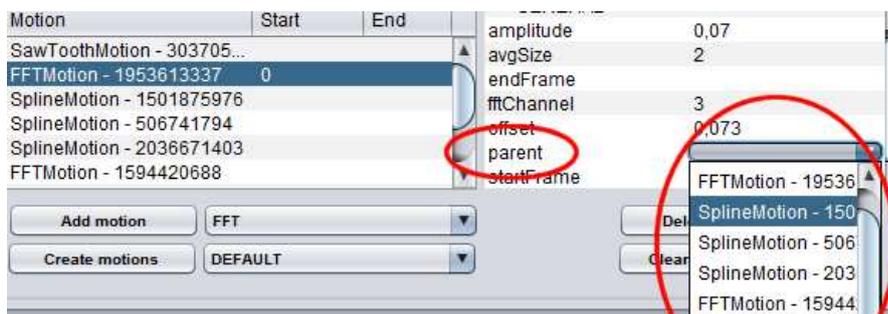
### 1.1.3 Motion links

To actually animate our fractal we just have to "connect" certain properties with certain motions. Any motion can be connected with any number of flame properties from any flame of the project.



### 1.1.4 Motion hierarchies

In certain cases it may neat to group motions together. E.g. to have a more "rough" motion (like a motion curve which describes a basic rotation) were a "small" motion (like "beat") is added to. To achieve this you may choose a parent for any motion.



A motion having a parent can not explicitly linked to a flame propertie (because it is already linked over its parent)

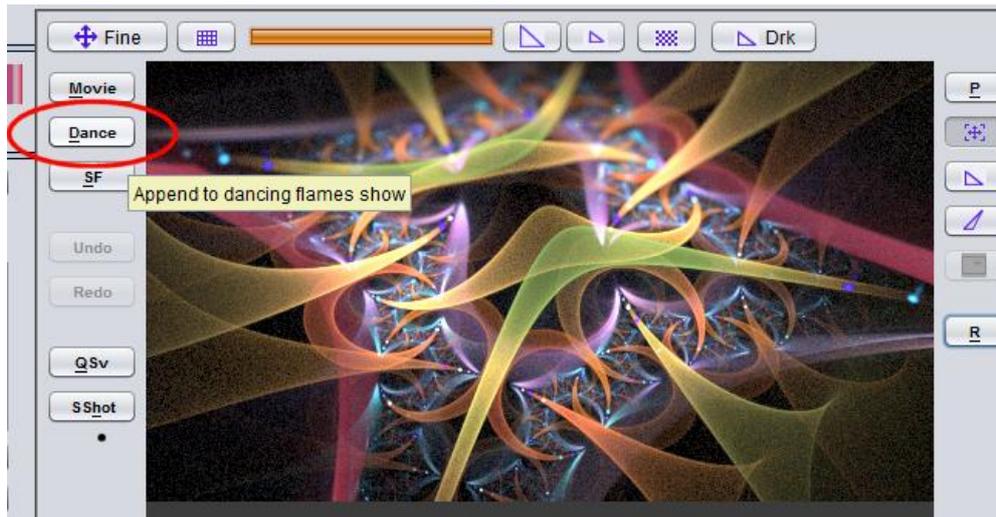
## 2 Basic documentation

In this chapter I will describe the basic usage of the concepts described in chapter 1, chapter 3 then will show some concrete examples.

To simplify things I will use the same flame example in the whole document. You can find the parameters of this flame in the appendix.

### 2.1 Getting started with a dancing-flames-project

The simplest way to start a dancing-flames-project is just to press the "Dance" button in the main editor.

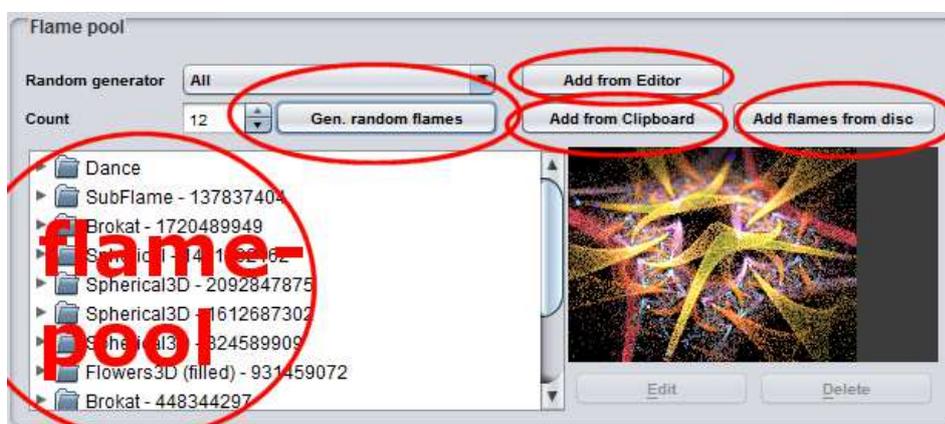


This will switch to the dancing-flame-module and append the current flame to the flame pool of your dancing-flame-project.

Of course, you can manually enter the dancing-flame-module and add any number of flames to the flame-pool

### 2.2 The flame-pool

The flame-pool is a collection of flames which belong to a dancing-flames-project and can all be animated into this project. Simple projects start with only one flame in the pool, but there is no limit and even if you have many flames in the pool, not all have to be used in the actual resulting video.



There a several ways to add flames to the pool:

- Generate a batch using a random generator
- import the currently load flame from the main editor
- add a flame from the clipboard
- load a batch from hard-disc

### 2.3 Adding sound (mp3)

Sound isn't actually required to create a dancing flame project, but the most projects will benefit from sound. Currently there can only be one mp3-file for the whole project (If you want to use more than one sound file, you have to merge them and load them as single file)

To add a sound file, just press the "Load sound" button and choose a mp3 file.



Please note that the loading may take a while as the sound is actually processed and analyzed in background.

### 2.4 Adjusting the projects FPS (frames per second)

The FPS setting of the project is one of the most important parameters, is affects how smooth your animation will be and how many frames (=single images) you will have to generate in order to create the movie. The more smooth your animation is, the more data has to be generated and the more large will be the resulting movie. In my personal experience a fps value of 25 (i. e. 25 images shown per 1 second of the animation) is a very good compromise.



### 2.5 Previewing the show

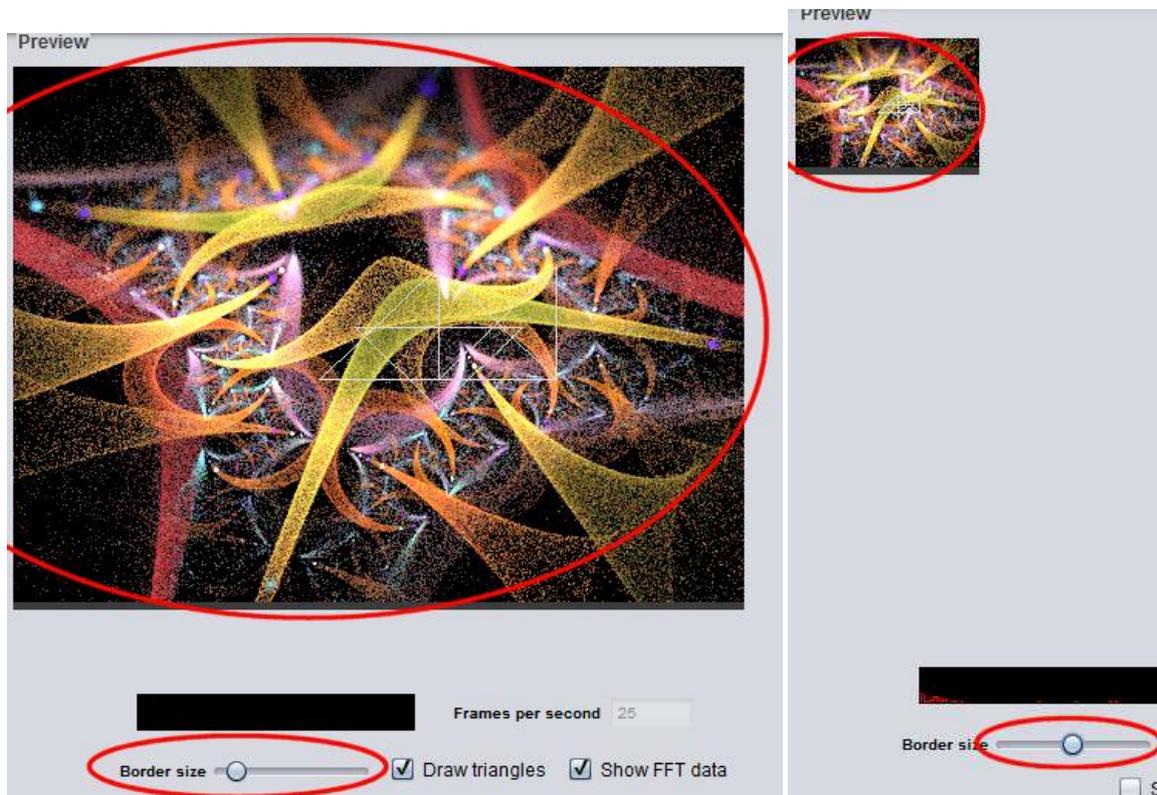
To preview the current animation just press the "Start Show" button. This will display a preview animation and play back the sound.



In many cases the preview will not be able to follow your FPS setting. This is no problem because the speed of the realtime-preview is not the speed at which the final animation will be created. The final animation is always created at the true fps setting, even at slow machines!

### 2.5.1 Adjusting the speed of the realtime-preview

You can improve the speed of the realtime-preview by decreasing its size. You can do this in real-time by modifying the slider of border size.

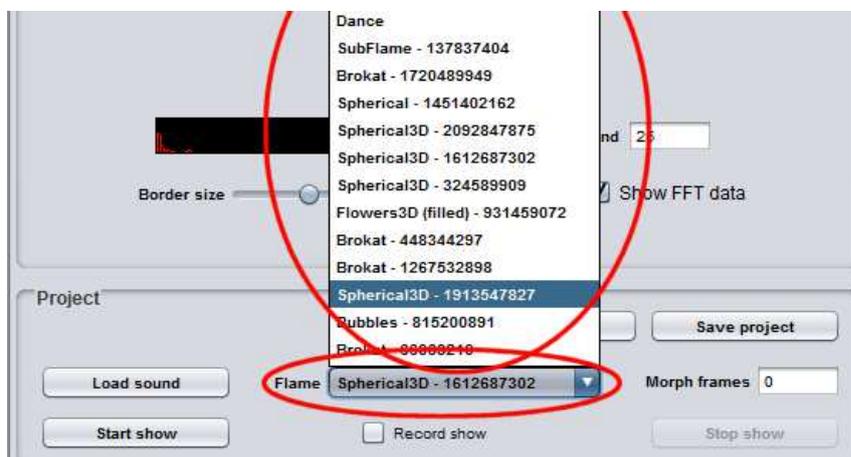


### 2.6 Switching between flames during the show

You may switch between the flames of your flame pool during the show all the time.

This is like a dj's job, and is currently the only interaction which is actually required to be done during the show. This is absolutely intended, and shall give your animation a more "spontaneous" or "looking-alive" behaviour. Just swap flames when you "feel" it :-)

To switch between flames just select them from the list-box in the project area during the show.

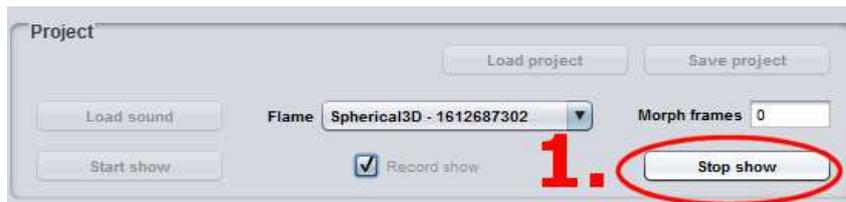


## 2.7 Recording a show

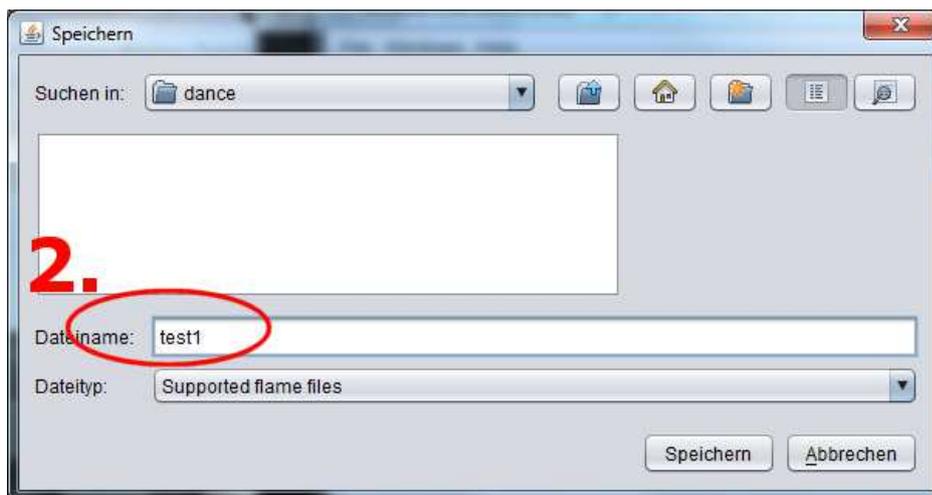
To record a show just enable the "Record show" checkbox and start the show. Now any user action (currently only switching between flames) is recorded accordingly to your fps setting.



After finishing the show (by pressing the "Stop show") button you will be prompted for a path.



Just choose a directory and specify a base-name for the flame files to be generated.



Now JWildfire recalculates the show again at accurate FPS setting and produces the flame-files which have to be rendered later to create the movie. Depending on the length of your show and the FPS setting this may take a while and produce lots of files.

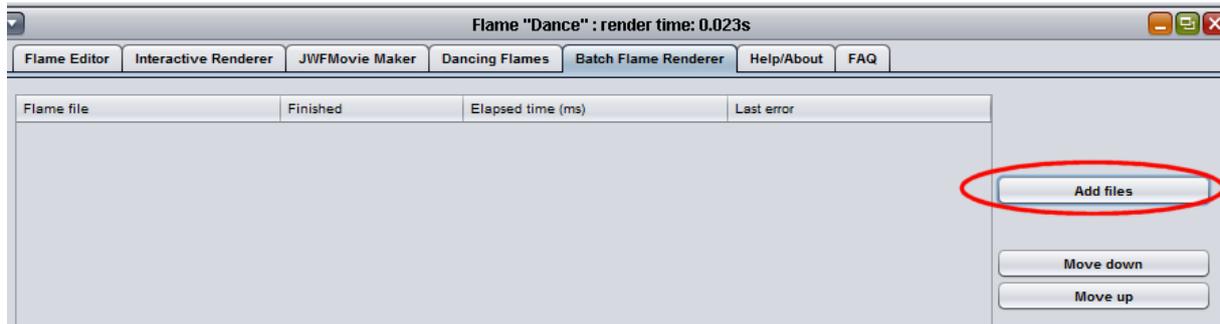
Please note that this phase is very important to have no hard interrupts (in order to create a really accurate timing), you should leave your computer alone with it, especially if you have a rather slow system.

## 2.8 Rendering the flames

As you may have noticed the project until now was free from any output resolution settings as we have only generated only flames. You can render them using any renderer using any resolution. The most simple way is of course to use the batch-renderer of JWildfire .

### 2.8.1 Using the batch-renderer of JWildfire

The batch renderer can render any number of flames at any resolution. To add files, just press the "Add files" button.



Note that you can in the dialog box which is opened next, select any number of flames AT ONCE. However, this does not work the same way on all platforms and unfortunately, don't know all of those platforms.

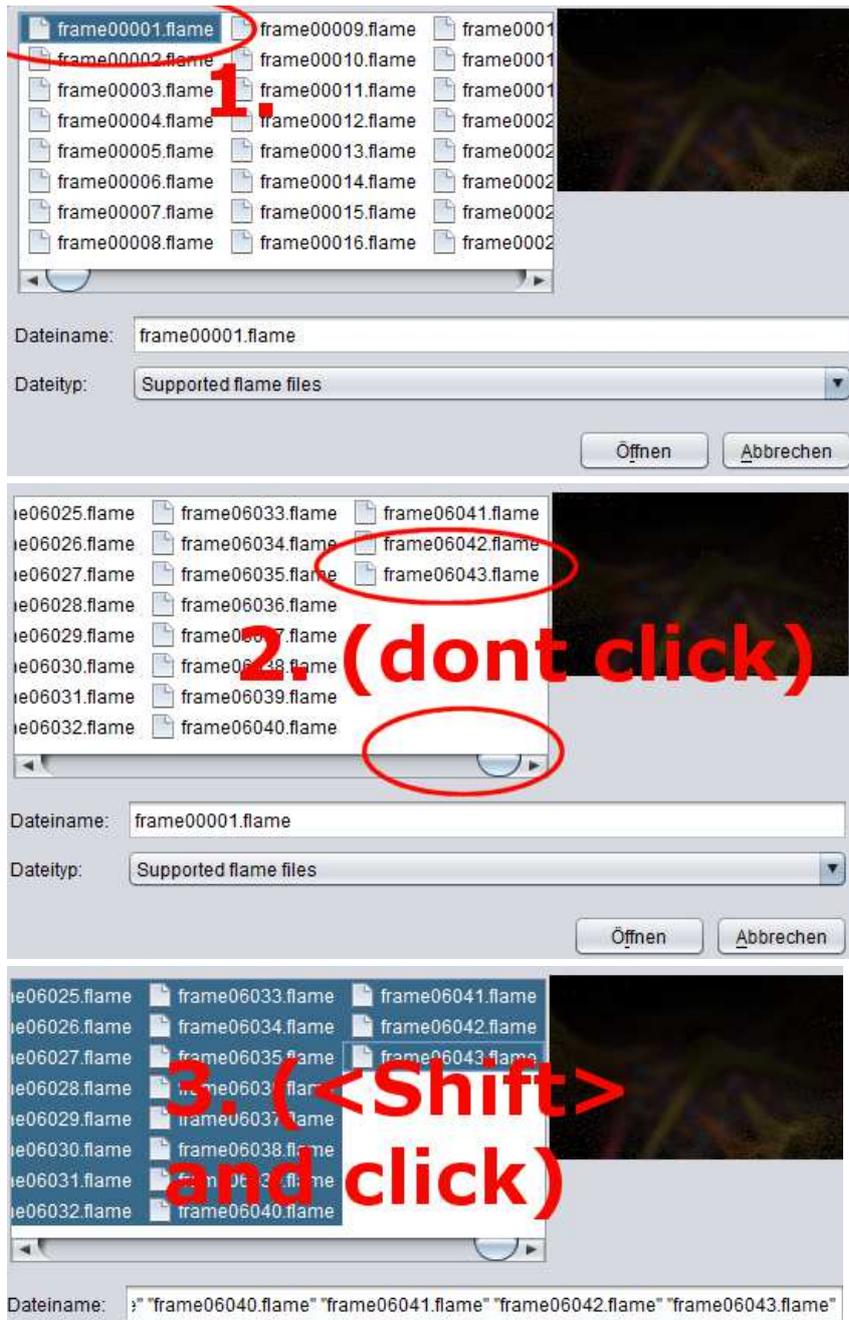
To select any flames inside a drawer there are two recommended ways.

#### 2.8.1.1 Loading a list of flames by using the "select all"-hotkey

- select any flame file with mouse
- choose the "select all" shortcut of your operating system ("`<Strg>+<A>`" on Windows)

#### 2.8.1.2 Loading a list of flames by using the "shift"-hotkey and the mouse

- select the first with mouse
- navigate to last flame, but not CLICK anything
- press and hold the SHIFT key
- click on the last flame



Finally click the "Open" (or however it may be labeled in your country)

### 2.8.1.3 Specifying the resolution and quality

Before starting the rendering of your flames please do not forget to specify both resolution and quality. For your first animations a resolution of 320x240 and a quality setting of "Low quality" (which is actually at this resolution is no low quality) should be a good starting starting-point.

### 3 Animation examples

Now as I have described both the basic concepts of the dancing-flames-module and the workflow related to create animations, I want to show some concrete examples.

To simplify things I will use the same flame example in the whole document. You can find the parameters of this flame in the appendix.

#### 3.1 A simple rotation

In this example we let a flame indefinitely rotate around the z-axis.

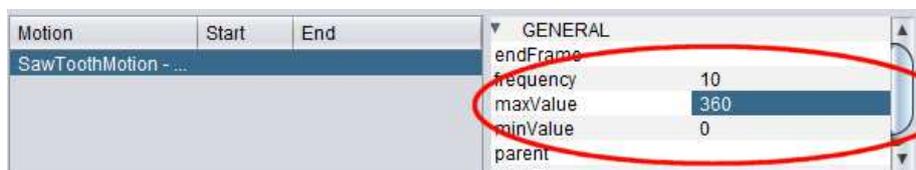
Start a new dancing-flames-project and add the example flame (or any flame of your choice) to the flame to the pool.



Add a motion of type "saw-tooth"



A "saw-tooth"-motion is raising a value from a specified minimum to specified maximum at a specified frequency. If we choose 0 as minimum and 360 as maximum we can use this motion as rotation angle (going from 0 to 360 degrees).



Leave the default frequency (which is rather fast) and let us link this motion to the camera yaw angle of the flame.

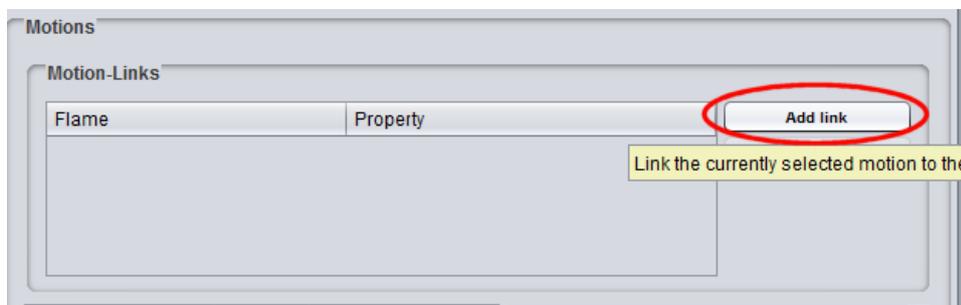
To do so we have to locate this property in the flame-tree-view. At first expand the folder of your flame at the top-level.



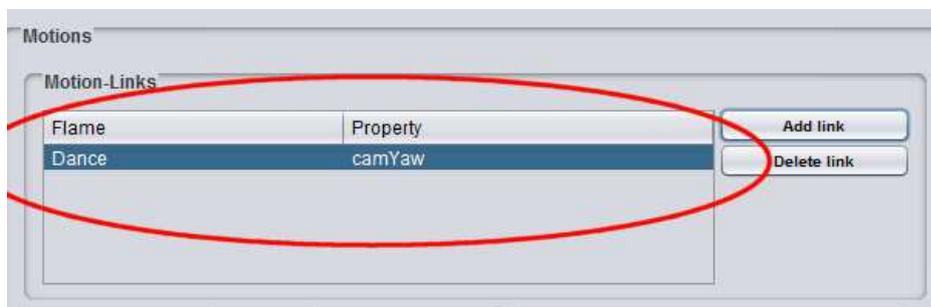
Locate the property by scrolling down.



Having both a property and a motion selected you now are to link them.



To do so, just press the "Add link" button. You now have attached a motion to the camera yaw angle!

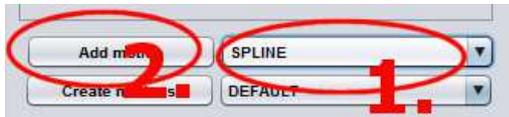


Congratulations, you just have created your first animation! Just press the "Start show"-button to watch a preview :-)

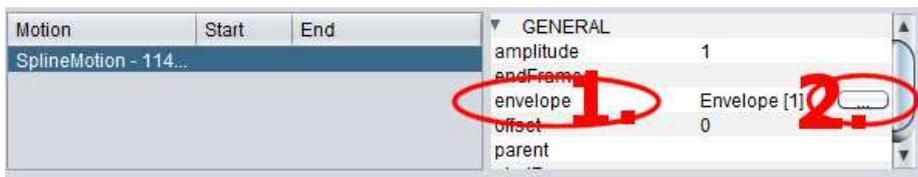
### 3.2 Animation by using a manually created spline-motion-curve

Now let's create some funny jumping motion by using the spline-editor. Please create a new project or clear all motions by using the "Clear all motions" button.

Add a motion of type "Spline".

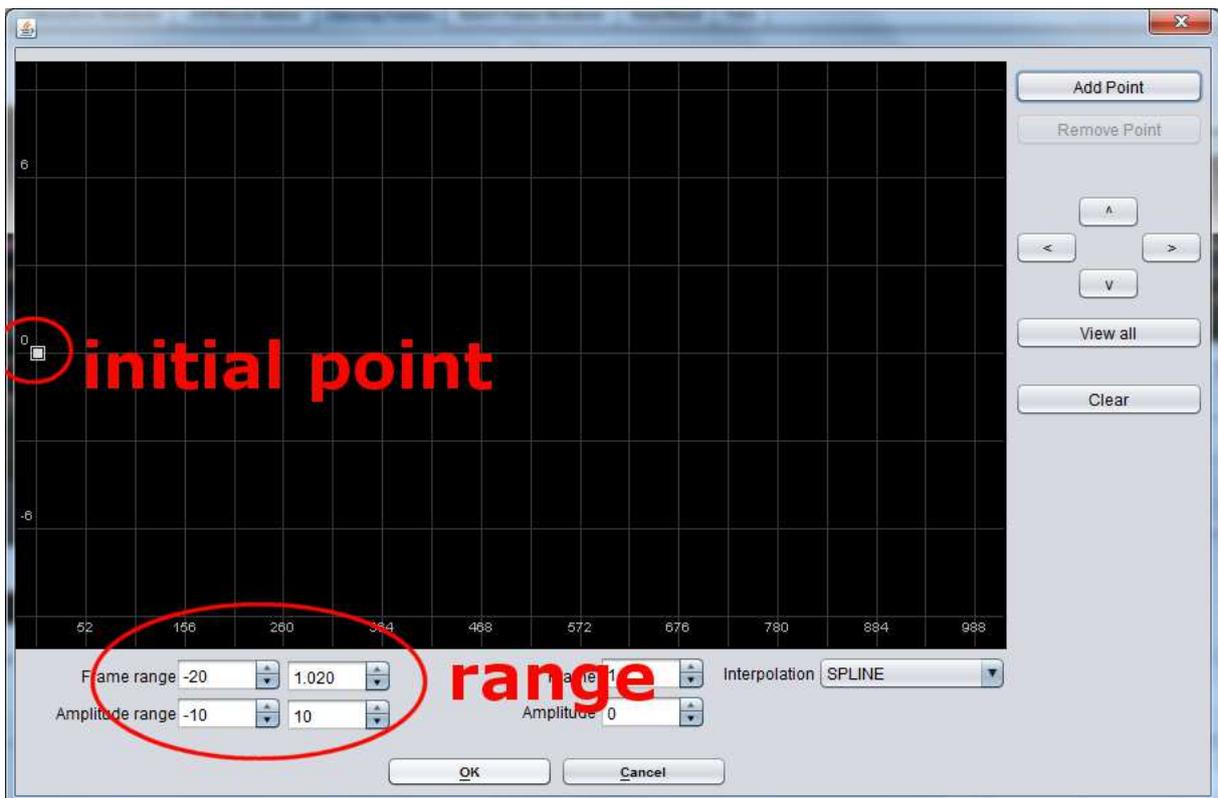


Enter the property table and locate the row with the "envelope" property. Double-click on it. A small button to invoke the spline-editor should appear.



Click the small button, this should invoke the spline-editor.

The spline-editor is used to manually create smooth motions curves (amplitude over time). Initially invoked the "curve" consist only of one point placed a frame 0, having an amplitude of 0.



But the most important thing here at the start is the frame range. The initial frame range is always initialized with 0...1000 where the display goes from -20...1020. Depending on your FPS setting this range corresponds to actual time (seconds).

Let's say you want to make a movie which has a duration of 30 seconds. With a FPS setting of 25 this makes 750 frames. So if you plan that the motion curve should go over the whole time, you should at first adjust the viewport accordingly.

Additionally you should think about the range of your amplitude. In our "jump"-example I will modify the amplitude between -0.5 and 0.5.

So let adjust the range of the motion curve as follows:

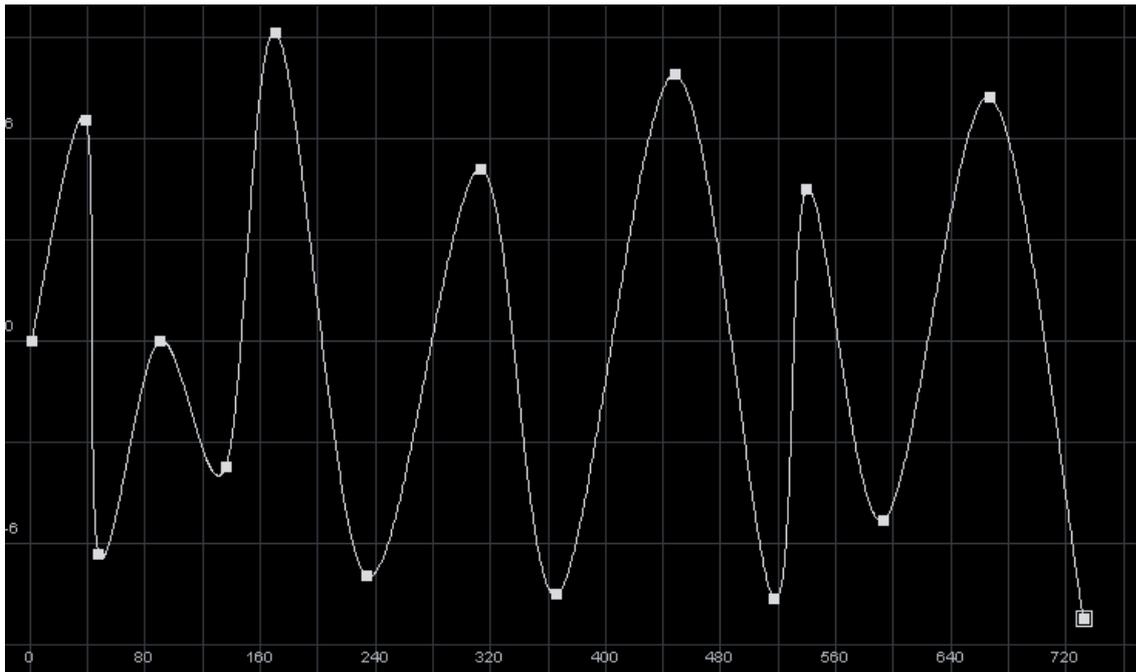
- frame range: -20...770
- amplitude: -0.6...0.6



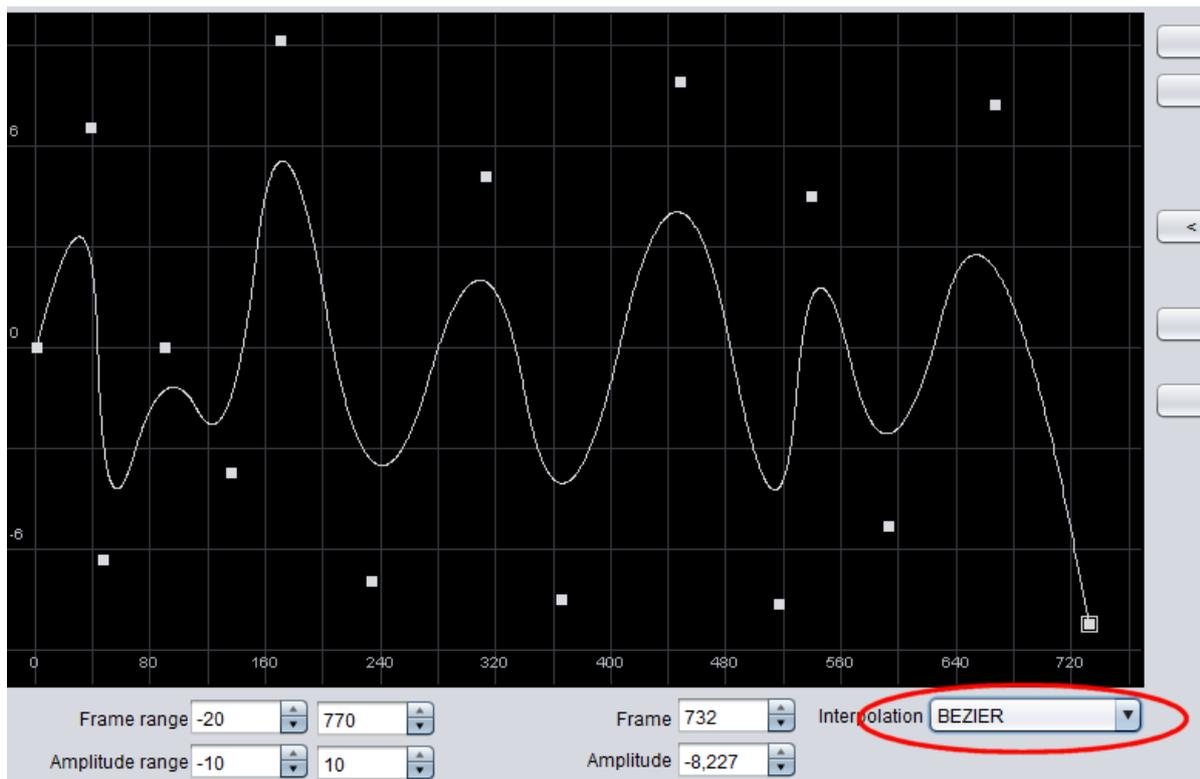
Now let's place our first point. Just press the "Add point" and then click at a point inside the black area



Add another point by repeating this, now you get a smooth curve. Repeat this until you get an interesting motion curve.

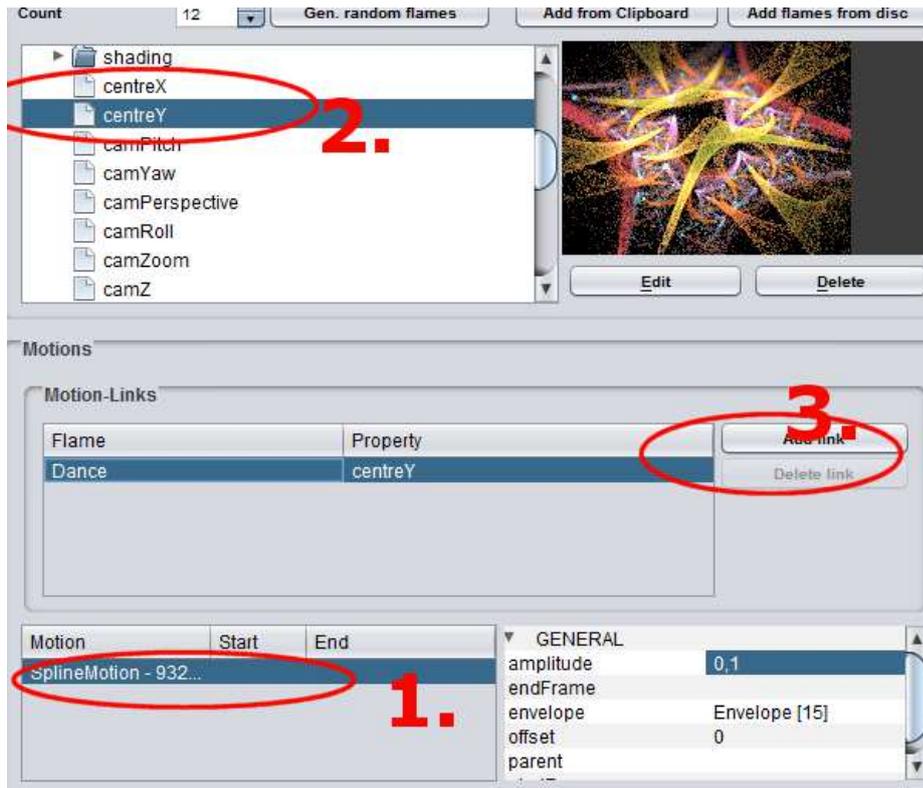


You can choose between different interpolation algorithm, for now let's choose "bezier"



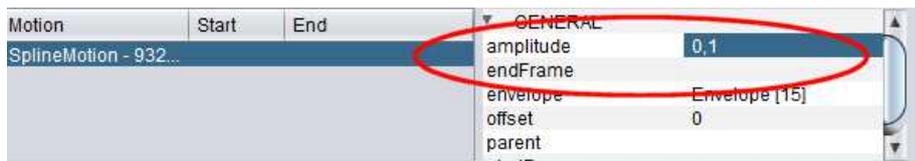
Confirm the motion curve by pressing the "Ok" button.

Now link this motion to the property "centreY" of your flame.



Start the show by pressing the "Start show" button. In the most cases you will now notice that the motion is "too heavy", i. e. the flame is jumping out of the viewport.

This is very intended at this stage ;-). You can both scale and shift you motion curves in the property table by modifying the parameters "amplitude" (default value: 1) and "offset" (default value: 0). To "scale down" our motion curve let's choose an amplitude of 0.1



If we now start the show again the "jumping" should occur inside the viewport.

Please note that this part of the tutorial was not intended to create a really overwhelming looking motion, I just wanted to briefly explain how to use the motion editor and therefore the example was intentionally chosen to be very simple and rather slow. Of course you can link weird motion curve to any weird flame property with any weird amplitude and speed ;-)

### 3.3 Finally: sound-synchronized motion :-)

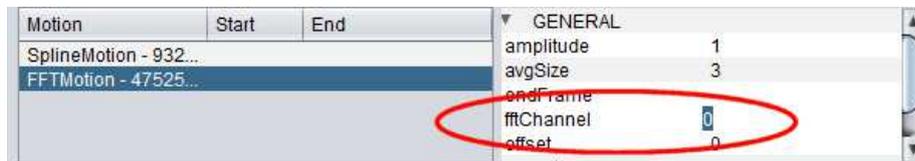
Until now we missed the "dancing"-part, i. e., sound-synchronized motion. But not without reason, animations only basing on sound are usually not looking very overwhelming. But if you add the "beat" to a well-animated movie it will likely become even better.

Technically, to add the "beat" to the motion, you just have to add a motion of type "FFT".



There are 64 frequency channels, which can be used as source for motion impulses. Depending on the song you use it may be more interesting to use more high or more low channels.

But if you are unsure it's also OK to leave the channel at the default value of 0 or to try low values in the range of 0..7.



To modify the intensity of the motion you have here also the parameters amplitude and offset. Usually you want only to change the amplitude (=intensity).

#### 3.3.1 Motion damping (smoothing)

Per default the motion is damped to achieve smoother motion. To decrease damping (i. e., to have a change in "beat" to have a change in motion in zero time, you can reduce the parameter "avgSize". To increase the effect of damping you can increase it. The actual behavior is dependant from the FPS setting, so just check it out

### 3.4 Using motion-hierarchies

Now we reached the most advanced topic here, the motion-hierarchies. Fortunately, their actual use is technically spoken the most simple one :-)

In the hierarchy there is always one parent which may have several children. Children may never be linked to flame properties, they are only linked to their parents.

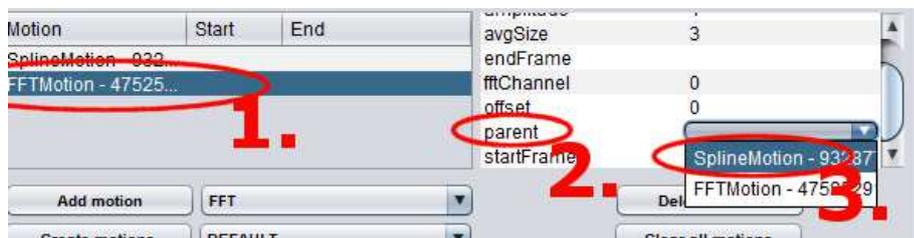
But of course, they affect the motion by adding "their part" to motion of their parent.

Imagine the "parent" as a heavy man, you could create his motion by draing a motion curve.

Now imagine the "child" as a little dog which is surrounding the man all the time the man walks.

The resulting motion is a global motion along a path (the motion of the man) with many small "glitches" (the dogs motion). A concrete example in the context of dancing flames could be a fixed rotation which has small modifications ("jiggles") by beat.

To achieve this, you can define any motion as child of any other motion. Just enter the properties table and locate the row "parent" to choose a parent.



Please note that such a child can't be directly linked to a flame property because it is only contributing to its parents motion (and so implicitly contribution to global motion)

*OK... so far with this tutorial, have fun to try your own things! :-)*

# Appendix: The flame used in all examples

```
<flame name="Dance" version="JWildfire 0.66 (09.03.2013)" size="510 287" center="0.0 0.0" scale="56.950971732286085" rotate="0.0" filter="0.0" filter_kernel="GAUSSIAN" quality="50.0"
background="0.0 0.0 0.0" bg_transparency="0" brightness="4.0" gamma="2.45" gamma_threshold="0.01" vibrancy="1.0" contrast="1.0" estimator_enabled="0" estimator_radius="0.7"
estimator_minimum="0.0" estimator_curve="0.36" estimator_kernel="GAUSSIAN" temporal_samples="1.0" cam_zoom="3.6" cam_pitch="0.645718232379019" cam_yaw="0.14477568036401176"
cam_persp="0.62" cam_xfocus="0.0" cam_yfocus="0.0" cam_zfocus="0.0" cam_rdimish="1.49" cam_rpos="0.0" cam_dof="0.53" cam_dof_area="0.5" cam_dof_exponent="2.0" preserve_z="1"
resolution_profile="320x240" shading_shading="FLAT" >

  <xform weight="3.47" color="0.2798220888694032" symmetry="0.9240126351833206" antialias_amount="0.75" antialias_radius="0.36" spherical3D_wf="1.0" spherical3D_wf_invert="0"
spherical3D_wf_exponent="2.0" coefs="6.123233995736766E-17 -1.0 1.0 6.123233995736766E-17 1.0 0.0" chaos="1.0 1.0 1.0 1.0 1.0" />

  <xform weight="3.8" color="0.6657599936199021" symmetry="0.976879645244216" antialias_amount="0.75" antialias_radius="0.36" spherical3D_wf="1.0" spherical3D_wf_invert="0"
spherical3D_wf_exponent="2.0" coefs="6.123233995736766E-17 -1.0 1.0 6.123233995736766E-17 0.0 0.0" chaos="1.0 1.0 1.0 1.0 1.0" />

  <xform weight="0.5" color="0.960611614813676" symmetry="-1.0" antialias_amount="0.75" antialias_radius="0.36" ztranslate="0.1238137631756768" pre_blur="0.4686273956513523"
spirograph="0.2369870627568247" spirograph_e="3.0" spirograph_b="2.0" spirograph_d="0.0" spirograph_tmin="1.0" spirograph_tmax="1.0" spirograph_ymin="1.0" spirograph_ymax="1.0"
spirograph_ci="0.0" spirograph_c2="0.0" coefs="1.0 0.0 0.0 1.0 0.0 0.0" post="1.0 0.0 0.0 5.0 0.0 0.0" chaos="1.0 1.0 1.0 1.0 1.0" />

  <xform weight="1.0" color="0.608814324496401" symmetry="-1.0" antialias_amount="0.75" antialias_radius="0.36" ztranslate="0.14886759358431037" cylinder_apo="0.073" pre_blur="6.17"
coefs="1.0 0.0 0.0 1.0 0.0 0.0" post="1.0 0.0 0.0 3.0 0.0 0.0" chaos="1.0 1.0 1.0 1.0 1.0" />

  <xform weight="0.1" color="0.36028762507955725" symmetry="-1.0" antialias_amount="0.75" antialias_radius="0.36" ztranslate="0.009209942145121778" gaussian_blur="0.005947966291207321"
coefs="1.0 0.0 0.0 1.0 0.0 0.0" post="1.0 0.0 0.0 1.0 0.3 0.0" chaos="1.0 1.0 1.0 1.0 1.0" />

  <finalxform weight="0.0" color="1.0" symmetry="0.5359245454231127" antialias_amount="0.75" antialias_radius="0.36" julia3Dz="0.6470433904619595" julia3Dz_power="2"
coefs="0.7071067811865476 -0.7071067811865475 0.7071067811865475 0.7071067811865476 0.0 0.25" post="1.0 0.0 0.0 1.0 0.0 -0.25" chaos="1.0 1.0 1.0 1.0 1.0" />

  <palette count="256" format="RGB" >

D95A71CB16DAE02DB8E54CDEBEA5C5E90CF04C42425E2A1979630D8D756E977534A4933E

A87A37A98AFAF6B49B1603DB4997ABA6F55C16917C49053C8AE7BCB513ECDB193D48F65

E0CD6AE546234C534C446939406C482976292F8046867F067E8D3F6A8E1A9094395A9569

978A124BA05D4DA21096A49865A88677AAA9AC887CAD6E90B03E5FB13739B30E79B92C

E3BC814BBE545FC05B87CB9C9CD8D7CCD3E48CD70AFCEA576D79265DD51E1CF584DE573

88E7A878F29FC0FBC12653573F5C5C24674F3376752A3A7D27908E5454972F4CA07362A5

4869A85A75AB3B9AAB418CAC53AEA03A7DAE3A5FB18598B2847DB53BB7A83C7EBA8B82BD

5A92C16FC6A72727C73309C993C5CA7A79D137ADD32E11D5A1B9D63A7FD7ADD5E3

34E9D3B8F1EF1FAAF7453643624F727A394A7D4953603F858A2C7C8D517A938F9376359E

A0629C836FA1854DA49907A65C1EA9AF8C9DB08687B264AAB7394FBB508C9F81BDC12565

C37A87C3B3C5C793C1C9B2B4CEAAB1CF7A81D282B9D63445D95A71CB16DAE02DB8E54CDE

EEA5C5E90CF04C42425E2A1979630D8D756E977534A4933EA87A37A98AFAF6B49B1603D

B4997ABA6F55C16917C49053C8AE7BCB513ECDB193D48F65E0CD6AE546234C534C446939

406C482976292F8046867F067E8D3F6A8E1A9094395A9569978A124BA05D4DA21096A498

65A88677AAA9AC887CAD6E90B03E5FB13739B30E79B92C3B8C14BBE545FC05B87CB9C

9CCD8D7CCD3E48CD70AFCEA576D79265DD51E1CF584DE57388E7A878F29FC0FBC1265357

3F5C5C24674F3376752A3A7D27908E5454972F4CA07362A54869A85A75AB3B9AAB418CAC

53AEA03A7DAE3A5FB18598B2847DB53BB7A83C7EBA8B82BD5A92C16FC6A72727C75674C7

3309C993C5CA7A79D137ADD32E11D5A1B9D63A7FD7ADD5E34E9D3B8F1EF1FAAF7453643

624F727A394A7D4953603F858A2C7C8D517A938F9376359EA0629C836FA1854DA49907A6

5C1EA9AF8C9DB08687B264AAB7394FBB508C9F81BDC12565C37A87C3B3C5C793C1C9B2B4

CEAAB1CF7A81D282B9D63445</palette>

</flame>
```