

Spinphony Demo

Mark Robert Nilsen
www.spinphony.com
markrobertnilsen@gmail.com

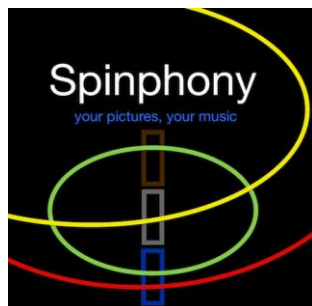


Figure 1. Spinphony Logo. Your Pictures, Your Music.

ABSTRACT

Spinphony is a program that turns pictures into music. The basis of this project was to create a program for iOS devices that would give anyone a tool to create original music. Spinphony uses the user's own pictures as input. Pixel data is turned into control codes that trigger Spinphony's sound engine. Some of the techniques used include note mapping and frequency quantization to coerce the data into a musical tonality and motif functions to give a rhythmic cohesiveness to the output. Additive and FM synthesis are the main sound generating synthesis types. The overall philosophy was to balance user interaction and internal algorithms as they relate to interpretation of the image-driven data stream. The result gives the program immersive interactive capabilities but also clearly image-driven musical output. Spinphony has succeeded in enabling non musicians and professional musicians to explore and create original music in a new way.

1. INTRODUCTION

Spinphony was recently parodied in The Onion. While that parody is not as funny as I hoped it does show that algorithmic composition is making its way into the mainstream enough that The Onion would choose to spend time 'reporting' it. You can find a copy of the parody video on the Spinphony web page.

Spinphony is available for free, world wide, through Apple's App Store. For those with iOS devices please search for 'Spinphony' in Apple's App Store or go to www.spinphony.com and proceed from the link there.

Copyright: © 2014 Mark Robert Nilsen. This is an open-access article distributed under terms of the [Creative Commons Attribution License 3.0 Unported](https://creativecommons.org/licenses/by/3.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

2. PHILOSOPHY

Spinphony is a digital version of a sound sculpture made for a show at HERE Art Center, NY, NY in 2002. The sculpture had a clear rotating disk and 3 oscillators. Anyone could draw on the disk. When the disk started spinning the image changed how the light fell onto photo sensitive resistors in the oscillator circuit of the sculpture. Short musical loops were created.

Spinphony has taken that idea and expanded on it. The idea is to give anyone the ability to explore music creatively and originally. Using the pictures and drawings from someone's photo library, Spinphony gives the user immediate and true sense of creating music from their own input. That is the thinking behind the phrase; Spinphony, your pictures, your music.

Another philosophy when developing Spinphony was to explore what sort of musical work needs to be done by the program when a picture, basically random input, is given to it to make music. And what can be left to the user's discretion through program interaction. An example of this is the motif function in Spinphony that gives rhythmic cohesion to Spinphony's output. Without the motif function the sounds are like a cat walking on a piano. Another obvious example of musical intervention from Spinphony is note mapping.

Every feature release of Spinphony is a rebalance of user interaction versus image-driven stimulus versus internal logic and how that balance affects the user experience of control versus the 'ah-ha!' of a new musical idea streaming from a picture.

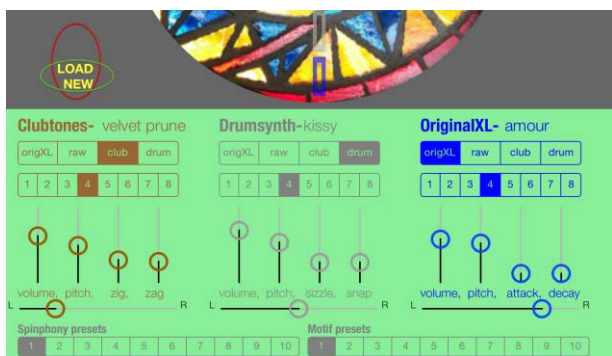


Figure 2. Screen detail from iPad. Shows how the controls are grouped appropriately for the screen size.

3. FEATURES

The features of Spiphony on the iPad, Figure 2, and the iPhone, Figure 3, are exactly the same. Because of screen size the controls are accessed differently. On the iPad there are 2 screens, one for note mapping functionality and the second to access synthesizer settings. On the iPhone 4 and 5 these functions are spread over several screens. The screens are accessed by swiping your finger across the control region.

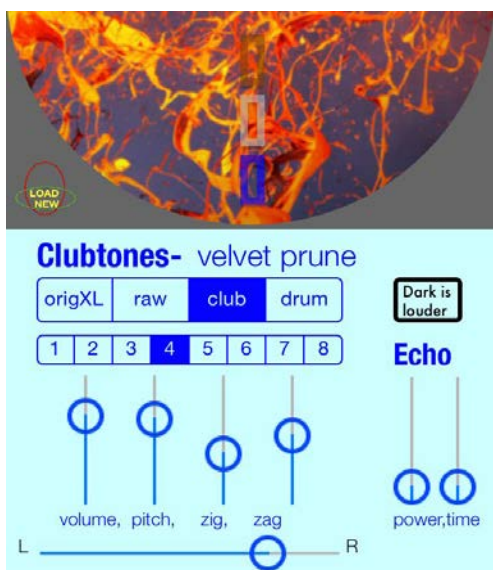


Figure 3. Screen detail from iPhone 5. Shows the lower part of the spinning image and synthesizer controls.

3.1 Image Loading and Control

Spiphony launches with a start image. To load a new image just press the 'load new' button to the left of the spinning image. You can then browse for an image through standard iOS controls. The image will load and immediately start spinning and making sound.

Swipe over the image with your finger to change speed and direction.

Use the stepper switch on a separate control screen to fine tune the speed.

3.2 Note Mapper

The note mapping function in Spiphony lets you coerce specific tonalities out of the stream of data coming from your image. By using the area of the program that looks like a keyboard, Figure 5 on the following page, and the selector switch below it, you can map Spiphony's note stream to, for example, a C Major chord, A Pentatonic scale or even Ab Diminished chord. It's pretty much all there. There is a free form selector '?' that doesn't coerce the notes to a key but lets them float around in an atonal way that can at times sound really cool.

Version 1.3 has a simple auto chord function. It changes the chord and chord color based on data from the image. In version 1.4 this feature is more developed and lets the user select chord progressions. Version 1.4 will be live for ICMC|SMC 2014.

3.3 Synthesizer Section, Changing Sounds, Manipulating Sound Parameters

There are 3 separate software synthesizers. They are color coded and each responds to a different region in the spinning image.

The controls have been set up to feel familiar to anyone who has used a synthesizer before. You can control volume, pitch range, and select from 32 basic sounds from 4 separate banks that each have 8 patches.

One of the controls, a button that changes from 'dark is louder' to 'light is louder' when pressed tells Spiphony how to interpret volume data. This toggle was added because it helps get better sounds from darker images or lighter images at the touch of a button. If you are ever stuck or frustrated with the sounds you are getting from the program try pressing this button.

There is a left and right pan on each synth. Spiphony sounds great with headphones and the ability to fine tune the stereo field was a must.

3.4 Effects and Motif Functions

Each synth has an echo effect. You can control how much echo there is and the time between echoes.

The motif function is the part of the program that ties things together rhythmically. Think of it as a grid. Music needs time cohesion or it sounds a bit, like I have said before, Fluffy on the Boesendorfer.

Figure 4 shows the motif control in Spiphony. The motif function 1X 2X with 20 as X. Can be thought of as an eighth note, then quarter note musical figure that will repeat. The value of 20 sets X to a size that makes the motif fast or short relative to the other X values in the other motifs. Use the first stepper to cycle through the motif functions and the second to change the value of X.

As simple as these functions are they open up a surprising range of possibilities.

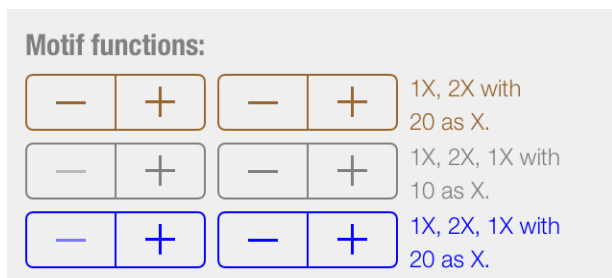


Figure 4. Motif function control

3.5 Saving Your Work

Spinphony has 2 preset banks. One for the synthesizer settings and one for the motif settings. This is so you can change the synth settings without changing the overall beat of what you have going and vice versa.

Whenever you change a slider or stepper in Spinphony the current active preset number changes to 'mod'. This tells you that you have modified a setting in 'x' preset. You can freely switch back and forth between presets and your settings from the previous preset will be saved.

To save this information to reload for another Spinphony session, use the 'Write Spinphony Disk' button. This takes all the data in Spinphony and writes it to a part of the picture outside of the circle that Spinphony uses for music. This Spinphony disk is saved to your photo library and can then be reloaded. You can even email these Spinphony files to other users. It should be noted that a Spinphony disk is a PNG file that uses part of the image area for data, see Figure 6. These files can be loaded into any image program just fine but if you add filters or change the image in anyway it will most likely corrupt the Spinphony data.

3.6 Inter-App Audio

Spinphony 1.4 will include Inter-App audio functionality and AudioBus integration. This feature allows Spinphony's output to be routed into recording or effects Apps that support this sort of interoperability.

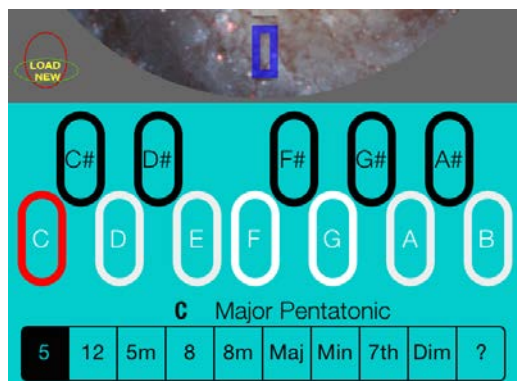


Figure 5. Note mapping screen showing mapping set to a C Major Pentatonic Scale.

4. HANDS ON DEMO

Sample Spinphony files are located at www.spinphony.com. Click the samples link. These sample files are already setup to produce some interesting music. Or you can load any picture from your photo library.

Don't worry about breaking anything or losing your work. If you come up with something you like, save it using Spinphony's save function. If you get stuck and cannot get a good sound, restart the program; Spinphony boots with a good sounding start image and presets. Anyone who doesn't know the program yet can get started experimenting from there. A good thing to try first is to listen to each of the presets and hear how things change.

Up to this point I have talked a lot about the Spinphony controls and how they change what you hear from the program. But, while browsing the sample Spinphony files you will notice that some of the examples are drawn images. Once you get comfortable with using Spinphony you will find that a lot of unique sounding music comes from drawing your own images. When you create images with varying areas of light and dark that would be difficult to capture when taking a snapshot you get results that have more variation than basic photos. Using Spinphony in this way returns to the original idea Spinphony evolved from; drawing a picture to create music.

Figures 6 and 7 show some drawn examples. The colors in figure 6 against the black creates some nice melodic leaps. Figure 7 shows an attempt at some deliberate triggering.

5. UNDER THE HOOD

Spinphony interprets your image in realtime. Spinphony's sound engine is responsive to all sorts of user interactions and of course the dynamically spinning image.

There were a few notable things I found while programming Spinphony that might be of interest to ICMC|SMC attendees who are interested in software development.



Figure 6. A hand drawn Spinphony image. Notice the data area (red) in the upper left corner.

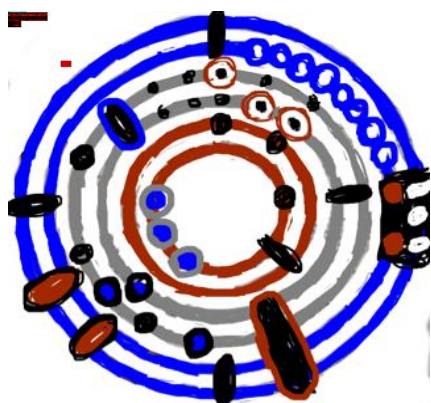


Figure 7. This image has a very deliberate placement of sound reaction areas.

5.1 Low Pass Filters

There is a lot of information/opinion on the net as to which filters are best. In my past job as a sound engineer I was always exploring what new companies had to offer in the line of filters. When working on Spinphony I spent some time looking around the net for some optimized sample code for a lowpass filter. I saw a lot of people discounting a moving average as a ‘bad low pass filter’. Well, as it turns out sometimes a bad low pass filter is all you need.

5.2 Poor Man’s Harmonics

When I was releasing version 1.1 I started rethinking my sound engine to get better sound quality. The first idea was to build up the standard sine, square, saw, triangle waveforms through additive synthesis. But calculating all those extra wave forms for a nice thick sawtooth wave bogged down the iPhone’s processor too much. After some trial and error I decided to just calculate $\pi/2$ (90 degrees) of the sine when building up some of the waveforms. Some cool sounds resulted from kludges like that.

5.3 The Old Tricks, Again

I started programming the Mac in 1987. Back then there was a program that used screen memory to help reduce the number of times you had to swap your floppy disk out when doing a floppy to floppy copy and you only had one drive. As the years went on, happily, memory became less of an issue. However I revisited that 25 year old screen image as data storage paradigm when coming up with the save function in Spinphony.

5.4 Visual Engagement

Seeing the spinning image engages on some level. Someone told me they like to watch pictures of their dog spin around. Maybe it even hypnotizes a little. It can take time to immerse a user in the creative experience and every little bit helps.

6. FEEDBACK, QUESTIONS

My involvement with computer music and sound, began in the late 80s and things have come a very long way since then. People are more engaged with listening to music in their daily lives than ever before. Algorithmic composition can now be presented to a larger audience. Spinphony is exploring a few of the possibilities. And I hope this demo inspires you all to explore your own ideas as well.

I would welcome any questions and feedback at this time. Thank you.

7. CONCLUSIONS

Processing of image data in a musical way can create engaging original musical compositions. The success of Spinphony today encourages the development of it further. However, too many features could overwhelm the user and disengage them. Spinphony needs to stay true to its image-data driven paradigm and develop processes that do the important work of sounding musical and original without burdening the user with too much tweaking of knobs.

Acknowledgments

I would like to thank the developer community at www.stackoverflow.com for the technical expertise they make available. Without them it would have taken me much longer to solve some of my problems. Likewise I would like to thank the developer community at Apple for making the tools available for Spinphony to become a reality.

Biography

Mark Nilsen, Figure 8, is a musician and visual artist, a Berklee College of Music alumni and extension school student of Harvard University and MIT. He has been appointed to MIT/Harvard University Research Lab of Electronics where he was an engineer for auditory research at Eaton Peabody Lab. He has written artificial intelligence engines for Philips Media and video compression software for Microsoft. Nilsen currently travels the world as a musician, artist and algorithmic composition evangelist mostly through promotion of Spinphony. He is also an instrument builder and founder of the experimental music group Trance Pop Loops.

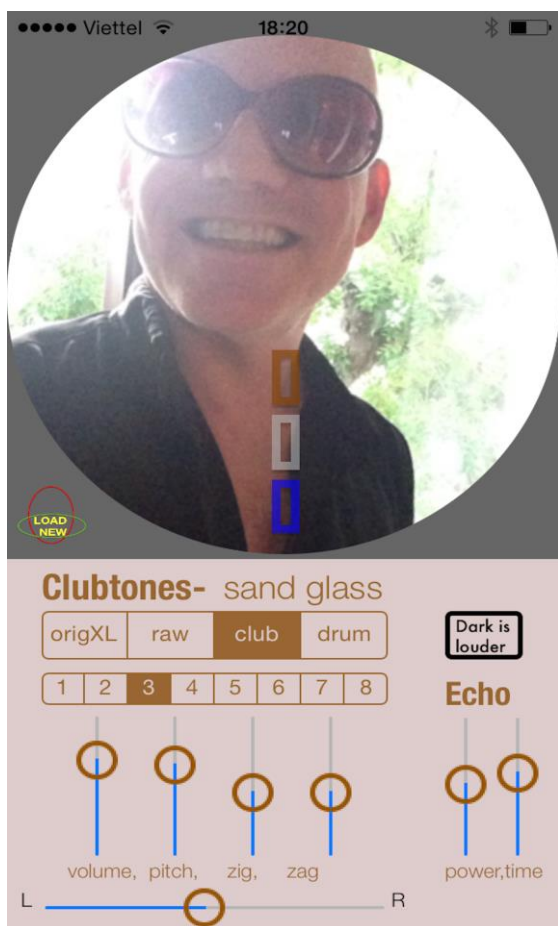


Figure 8. The developer in his program.