



Zachary Lieberman and Theodore Watson
interviewed by Friedrich Kirschner

openFrameworks

Can you tell us a little bit about yourself and how you got into all this?

ZL: I'm an artist and researcher who has been working for the last 6 years creating interactive projects that deal with the nature of how people communicate and making things come to life. I started out as a fine artist and got into software through animation, during the dot-com era when folks were coding creative visuals in flash. The moment I had coded my first rectangle bouncing off the monitor's edges, I was hooked. From that point on I dedicated my time to making interactive projects and working on my own and together with Golan Levin created a variety of projects, many of which have been shown here at Ars Electronica. We try to make stuff that is immediately understandable, playful, expressive and absurd.

TW: I am Theodore Watson and I work in a strange field of designing/making interactive environments and technology. This usually involves adding a "layer" onto a space or object that normally wouldn't be there but that lets you do fun and amazing things. I have worked on making giant vinyl records which you have to run around the surface of to play, the Graffiti Research Lab's Laser Tag system which lets you write a message or "tag" across huge buildings with a laser pointer, an interactive ecosystem for children and an invisible 3D sonic environment which lets you leave messages floating in space with your voice. I try to make things that seem impossible and transform the way people play with their environment.



***openFrameworks* seems a bit hard to describe—can you break it down for non-programming people?**

ZL: We describe *openFrameworks* as a creative toolkit for programming. It's basically a system that makes it easy to get started coding and to get into working with software without knowing too much. It's not for beginners, but it's designed to be simple and lightweight. When we say framework, we like the visual metaphor of scaffolding, the idea of a kind of lightweight structure that you can use to build complex and interesting works with. Also it's the idea that we are artists developing tools for other artists. As we work, we take ideas that we learn and try to bring them into *openFrameworks* so that others can use them. It makes us better coders and it also means we can have a larger impact than just making a project with our names on it. The goal is to take seriously this notion of *art making as research* and this is one way of publishing the results of that research.

And for the programming people that use library packages day in day out, what's special about it?

ZL: It's designed to be simple, lightweight and easy to extend. Also, since we are artists first and foremost the goal is really to help people like ourselves. Other libraries have other target audiences—application developers, 3d game programmers, etc, and with us, the goal is someone interested in exploring audiovisual concepts in an intuitive and powerful manner. We are simple, but we provide low-level access to objects so the folks can play computationally.

TW: It is also cross platform (yes Linux too—thanks Arturo) and the code is really portable! We have designed it so that the software you make runs the same—no matter which platform you are on. I developed the *Laser Tag* project on a Windows machine. Once it was done 15 minutes later there was a version for Macintosh and once I had setup Linux on my laptop there was a Linux version too. The project was a large project and I was really amazed when I saw the software run exactly the same on Mac and Linux—with no changes to the code.

Where do you see the artistic value of *openFrameworks*?

ZL: I think when people have better tools they make better projects. I think the works that are coming out of the *openFrameworks* community are playful and expressive and varied, and it's a good sign that we are doing something right.

Also, *openFrameworks* is helping a new generation of artists make works that are leaving the screen. From people using lasers to draw on buildings to noodles that swim off the monitor onto your hand, to submersing people in black ink in stages to generate a 3d model of their bodies, OF projects seem to be on a forefront of a new line of research in interaction design—real time systems that alter peoples' relationship with physical space.

What do you think is the reason that so many people respond to *openFrameworks*?

ZL: People seem to respond to open source, especially when it's easy to get started. Also, it's a powerful tool in a simple shell, so it feels pretty good. Since it's transparent, it's easy for people to get involved in fixing the software, and obviously, in criticizing it too. That kind of feedback and dialogue is important. We have an active community of folks developing/modifying/sharing and that makes the code better. Another point, it sounds really silly to say, is that questions get answered quickly. That makes a big difference when you are just getting started with coding and struggling with complex error messages or how to solve a problem.

You use *openFrameworks* in your own work as well?

TW: Almost all my projects are made with *openFrameworks*. I don't use it out of a sense of habit or loyalty but just because I can't think of another tool or software that wouldn't limit me in one way or another. The tools people use to make work can often shade its final output. I like using *OpenFrameworks* because I know that if I think it, I can make it and I don't feel like it is forcing me to work in a certain way. *openFrameworks* also gets me through all the tedious parts of programming so that I can put my time and energy into writing the code that really makes a project unique. A great aspect of developing *openFrameworks* and also using it for projects is that you are constantly testing the code and able to see how well it works in practice. I don't think *openFrameworks* would be where it is today if Zach and I weren't constantly using it in our own work.

ZL: And that back and forth also informs our work—it's gives us a flexibility for expression that we didn't have before because much of the hard stuff is taken care of. It makes making things quicker and much more fun.

How difficult is it to keep the balance between providing an easy entry for newcomers and still having all the functionality that both of you need for your work?

TW: I don't think it is that difficult actually. The way it is structured is that the core *openFrameworks* library caters to the most common tasks—things the people use a lot (image, type, video, sound input/output etc). We then have an “add-on” system where people can write mini-libraries for *openFrameworks* that aren't available by default but can be easily added to a project. These add-ons tend to deal with more advanced functionality or for solving very specific problems, like tracking the location of a finger. The nice thing is that people can start off very simply, but if they need that extra functionality it is there. For my work I find that with the core library and the add-ons it covers a lot of what I need. The rest of the code I have to write is often just specific to the project in hand. However if I am writing code that seems like it would be use-

ful for many different projects, I try and develop it as an add-on and share it with the community. For example I was working on a poster project for children and I needed a way to export algorithmically generated designs from *openFrameworks* as vector graphics, so I made the *ofxVectorGraphics* add-on that lets you save a screen grab as an EPS file. Several people have already used it in their projects.

Why do you give out your tools for free?

TW: I get asked this a lot—usually after presentations. People can't seem to understand the logic of it. I can understand their confusion, but for Zach and I it actually makes a lot of sense that *openFrameworks* is open source and free to use. Working on *openFrameworks* is not our fulltime job—it is something we work on as we are developing our own work and as it is a tool that we use, working on *openFrameworks* means that we are improving our tools. We could of course charge other people for these tools, but we prefer that other people can use them for free and help make them better. Making the software free opens it up for anyone to use and in turn has resulted in a large and active community. At the moment within the community there is a lot of development that Zach and I have very little to do with—which is great for us! If *OpenFrameworks* was developed like a traditional software business I don't think it would be anywhere as rich or broad as it is today. It would also be a real job and no way near as fun to work on!

At Ars Electronica this year, you are presenting the *openFrameworks Lab*, can you tell us what the goals are?

ZL: The *openFrameworks Lab* idea is really simple—we will gather a group of really talented *openFrameworks* folks together in one space and make art projects which the audience suggests to us. Since the *openFrameworks* project is really about fostering a creative community, we like the idea of trying to get people under one roof. And then, instead of just working on projects we would normally work on, we work on what the audience suggests to us. The art is created live during the festival, and we take the projects out into Linz. We have no idea, and very few preconceptions of what we'll be making, but we know we'll be making it together as a laboratory, and together with the audience of this year's festival. That unknown is exciting.

What was the nicest thing you heard about *openFrameworks*?

ZL: I demoed *openFrameworks* for Toshio Iwai, who is one of my all-time heroes, when he was in town for the launch of the Tenori-On. He said that we were in the same business of “creating operating systems for artists.” I love that idea, and for me it was a huge compliment.

TW: Most of the praise I hear is either indirectly through websites and blog posts or actually more directed towards the work made with *openFrameworks*. When I see projects like *Contact* from UnitedVisualArtists or Jonathan Harris's new installation *I Want You to Want Me* and see people's reaction to it, there is a great feeling from the fact that people find *openFrameworks* useful and are making amazing work with it.

What was the weirdest thing you heard about *openFrameworks*?

ZL: At Sonar, I bumped into a Chilean artist and his filmmaker girlfriend. He said “Thank you so much for *openFrameworks*!” and his girlfriend slapped him on the back of his head with a laugh and said, “Damn you! He spends all his time coding now!”

