# PERFORMING PROCESS:
# THE ARTIST STUDIO AS INTERACTIVE ART

LUIS BLACKALLER

Thesis proposal for the degree of Master of Science
at the Massachusetts Institute of Technology
Fall 2007

Thesis Advisor_____

John Maeda
Associate Director of Research, The Media Lab
E. Rudge and Nancy Allen Professor
of Media Arts & Sciences

Thesis Reader_____

Casey Reas

Thesis Reader_____

Nick Montfort

*Date*: October 26, 2007.

Abstract. The interest of this work is to explore the intersection between interactive art and digital graphics. I will describe the design of a system where the creation process will be shared live by the artist in an interactive online space for participation. This will potentially open a conversation between artist and audience that will shed new light on how we can learn, understand and communicate digital graphics.

## Contents

## Introduction

Graphics and interaction were coupled together long before the digital era. If interaction is understood as a kind of action that occurs when two or more objects have an effect upon each other, and graphics as the collection of all two dimensional visual representations, it can be said that the simple acts of drawing and writing are both interactive graphics, acting upon their creators when they were decoded back by them, illustrating the feedback nature of artistic creation. Norbert Weiner [20] explored a similar idea when he defined cybernetics as the study of control and communication in the animal and the machine.

However, the act of drawing and the finished drawing are two different things, and graphic art has been traditionally considered to be the finished drawing rather than the process of drawing it. Because of these, the participation of audiences in the appreciation of art has been a history of contemplation against an art that was mysteriously created by a mythical creature called the artist.

In the beginning of the 20th century, Artists like Marcel Duchamp [22] and the Futurists were trying to understand the impact of the rapidly evolving technologies over human life, and in some cases brought a reflected notion of interaction to the visual arts, where the audience would be tantalized by an art that would invite them to use it, even if the social conventions of the gallery, or the museum, would immediately repress them from attempting to break the sacred separation between art and people (Bicycle Wheel by Marcel Duchamp, 1913). Only later instances of experimental theatre, and performative art movements like Fluxus and the Situationists, would openly suggest an art that would give its audience the power to decide when or how they would interact with it, and an opportunity to become a part of it. This kind of tension between the sacredness of art and the accessibility of other media has evolved into the current state of crisis in the art system, that has tried to increase its influence by shifting its domain towards the realm of mass

media, but losing as a consequence a clear notion of what actually constitutes ART, a volatile term that has come to stand for whatever a group of collectors or curators choose to promote.

This is the scenario where the digital arts were born. On one hand, heavily abused by industrial media production, digital resources have been crippled by the representation of tools that are merely used to make a non-digital process cheaper and faster. On the other hand, the interactive potential of the digital medium has inspired the awareness of a strong international community that has pushed the boundaries of digital art outside of its limits, helping deliver the complex digital culture we experience today. Myron Krueger [10] [23], a pioneer of virtual and artificial reality, used to say say that interactivity should be raised to the level of an art form, as opposed to just making art that happened to be interactive. In the MIT Media Lab, student researchers like Bob Sabiston [19] [18] (1991), Golan Levin [12] [11] (2000) and Casey Reas [14] [13] (2001) among others, have made significant contributions to the field.

However, learning how to manipulate digital media is a challenging process. The artist usually depends on technologies that are not available everywhere or might become obsolete next year, complicating distribution and conservation even more than with traditional materials. In recent years, a family of systems led by Processing [15] [16] (VVVV [5], Max MSP [2], Quartz Composer [4], etc), have been developed as an effort to simplify the process of prototyping interactive digital systems, and make it easier to distribute and run the content across different technologies. Part of the success of Processing, for example, comes from the option to experience versions of the work of the artist everywhere inside a webpage, opening a space online that rapidly incorporated blogs and forums as the standard to build communities where digital artists or designers could learn from each other. Over time, other options were considered when designing this kind of spaces, perhaps looking

to stay closer to the nature of the content they represented. SodaPlay [7] lets its members program interactive animations by visually manipulating the parameters of a physics spring-mass simulation, and save the results on a social networking site for sharing. OpenCode [9] Makes it possible to write and run Processing code from a webpage, automating and hiding in the server side the process of compiling to an applet an embedding it on the page.

## Problem

There are three problems that motivate this work in its relation to digital art and contemporary culture. The first problem is the aesthetics, or appreciation of digital art. How can be digital art understood? In order to explore this question, two related problems follow immediately: the poetics, or process of creating digital art, that looks to answer the question how is it made, and the problem of accessibility and distribution, that looks to answer for the question of how to get or experience it.

In an effort to attack this problems I plan to narrow the scope of this thesis by focusing only on the expression of art through interactive digital graphics, and limiting the materials of the artist to code. Inspired by the romantic vision of an artist sharing working hours in his studio with a group of visitors, I propose to build a system where an artist can gather an audience online and interact with them while iterating through the sequence of modifications made on the code that generates and runs the art. I will use the system myself taking the role of the artist, and exposing my process of creation to the feedback of a remotely located audience.

## System

The proposed design must include a space for the artist to write and run graphics code, and the necessary implementation to dispatch data (such as code, images and other suitable content) to the server at will, distributing it along a group of participants online, and giving them means to send feedback to the artist.

An important component of this project that is NOT included in the design of the system is the artwork itself, whose development will be broadcasted in a number of sessions to test the system with a real audience. The system will be my tool to communicate my own art.

## Resources

The studio for the artist will be created on top of E15 and oGFx, two projects recently developed in the PLW. oGFx is the graphics engine that powers E15, an interactive multidimensional environment for exploring the web beyond the browser. I will build the functionalities required by the characteristics of the artwork, and for communication with external resources that will deliver interaction to the rest of the participants. The core technologies underneath E15 and oGFx are the Cocoa framework from Apple (Objective C, CoreImage and Quarts2D), OpenGl (and GLSL), and Python.

Broadcasting and feedback will be orchestrated by a web application built on Ruby on Rails, and every aspect of the process and communication will be stored in a database for later examination.

## Contribution

This work will help to explore digital media when liberated from the constraints imposed by its current state, and contribute to validate code as a full featured mean of artistic expression that must find its rightful place into the contemporary art discourse, as one of the richest cultural constructs of the 20th century (code is the true new literacy that fuels new media). It will offer an alternative option to create and experience digital art, and the development process will help me elaborate more precise ideas towards the poetics and aesthetics of digital art as interactive graphics.

## Evaluation

A number of participants will be invited to use the system with me in a number of running sessions. The data gathered from these sessions will be examined in a qualitative way, looking for inspiration to think about the following questions: How does the art change when the audience can interrupt the process? Is the art in the performance of the process? Can the process be appreciated as the finished piece?

## Figures

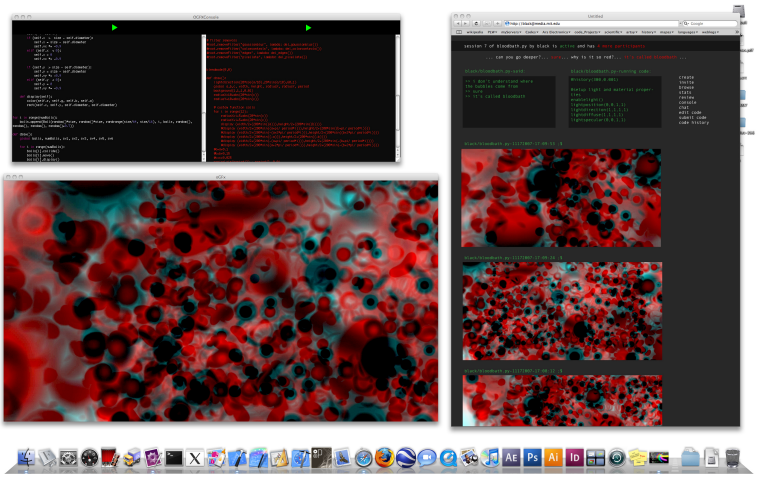

Figure 1. System Components: Scripting console, interactive graphics view and web application. Any participant can interact with the artist through chat, code exchange and image appreciation, experiencing live data coming from the artistic process. If the participants have access to an instance of E15 or oGFx they can run their own modifications and submit them to the artist for consideration.

```
#  bloodbath.py
#  OpenGFX
#
#  Created by blackaller on 10/24/07.
#  Copyright (c) 2007 MITPLW. All rights reserved.
#

#updatefromweb()
#broadcastcode()
#broadcastview()
publishframe()

#history(300,0.001)

#setup light and material properties
enablelight()
```

Figure 2. System Components: Scripting console. All the communication functionality will be available to the artist as a library of customizable methods that will be embedded in the same script that runs the interactive graphics.

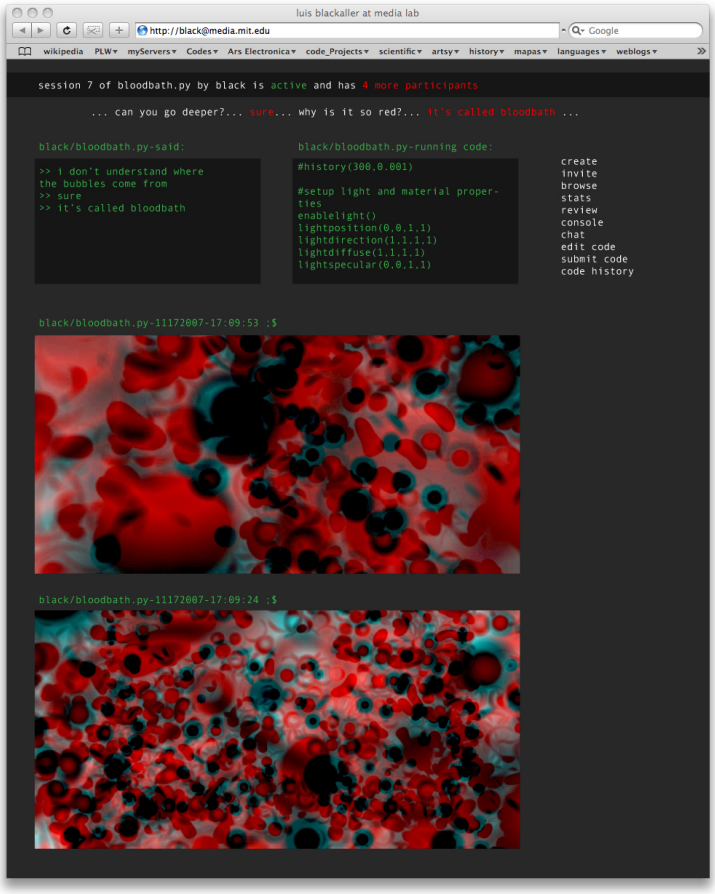


FIGURE 3. System Components: Web application.

FIGURE 4. Interaction. The picture shows the manipulation of an early experimental script created using the oGFx [6] graphics engine developed by Kyle Buza [8] and me over the summer of 2007. All figures featured in this paper are the results of testing the oGFx engine, and inspired us to create a language to control the expressive power of these interactions.

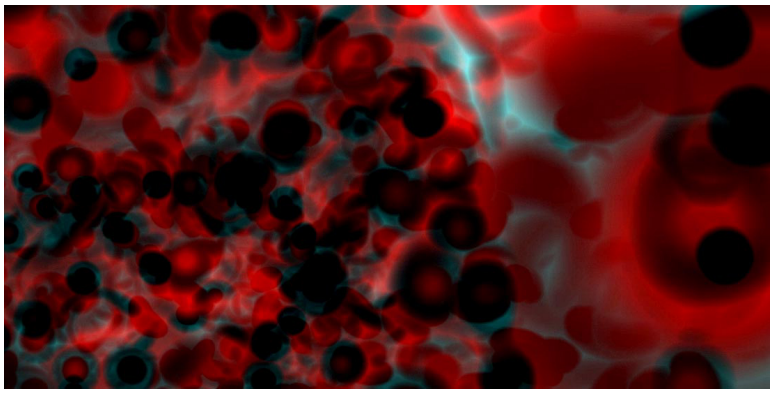

FIGURE 5. Art Experiment 1: Blood. Interactive control over the different blending modes available in computer graphics facilitates an entertaining way to understand pixel arithmetics through play, and delivers interesting immersive experiences.

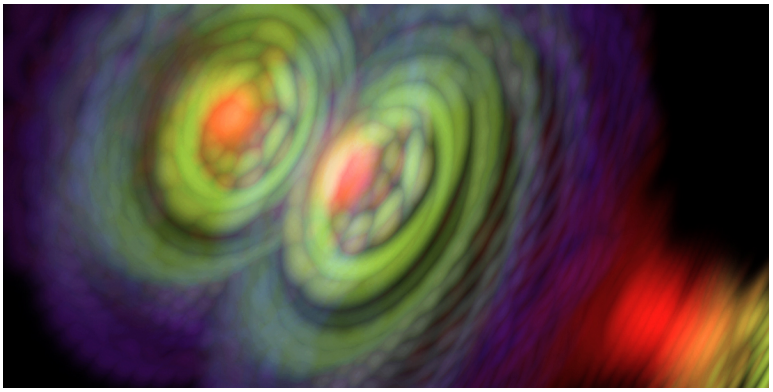

FIGURE 6. Art Experiment 3: Bomb. A simple implementation of a dynamic lighting model paints a colorful effect over the history of a blurred animated texture.

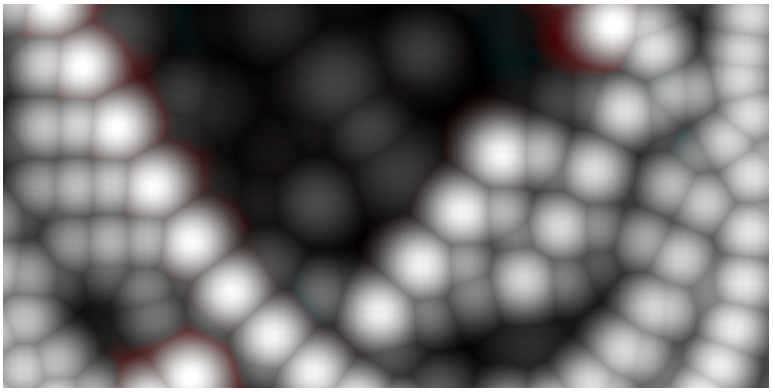

FIGURE 7. Art Experiment 4: Tissue. A simple change in color and blending and the texture that generated the blood environment creates a tissue-like tunnel structure in space.

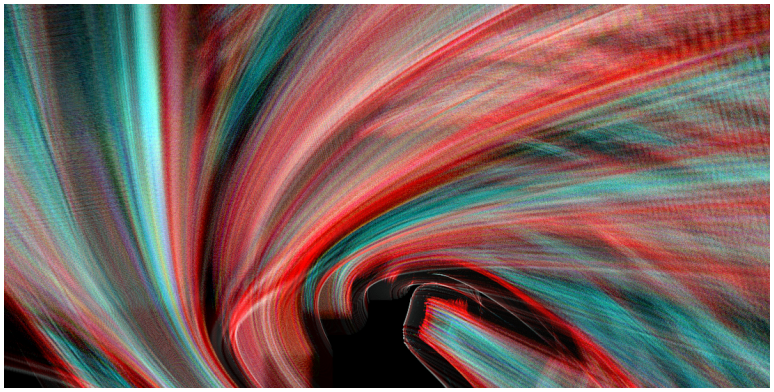

FIGURE 8. Art Experiment 5: Fiber. OpenGL 2.0 [3] features the capability to dynamically change shaders and shader parameters through GLSL [17] [1]. Because shaders run inside the GPU, it is now possible to interact with interesting visual effects that used to belong to the static realm of ray tracing.

FIGURE 9. Art Experiment 6: Video spaces. Futurist in spirit, an interval of 600 frames of the film Tron is extruded over the 3D space. This visual exercise has been explored multiple times since Etienne Jules Marey's [21] early photographic experiments, but it has been not until recently that graphics processors are powerful enough to experience the spatial projection of time in motion.

## Timeline

October 2007: Concept refinement. Proposal. Research and Documentation.

November 2007: Design, early prototypes. Research and Documentation.

December 2007: Thesis background, introduction and approach.

January 2008: Final stages of development. User testing.

February 2008: Analysis. Iteration, more testing. Thesis design and implementation sections.

March 2008: Thesis evaluation and conclusion sections, proofreading.

May 2008: Finished thesis.

## References

[1] Glsl. [http://www.opengl.org/documentation/glsl/].
[2] Max msp. [http://www.cycling74.com/].
[3] Opengl. [http://opengl.org/].
[4] Quartz composer. [http://www.quartzcompositions.com/].
[5] Vvvv. [http://vvvv.org/].
[6] Luis E. Blackaller and Kyle M. Buza. ogfx. [http://web.media.mit.edu/ black/ogfx/].
[7] Ed Burton. Sodaplay. [http://sodaplay.com/].
[8] Kyle Buza. [http://plw.media.mit.edu/people/buza/].
[9] Kyle M. Buza and Takashi Okamoto. Opencode. [http://opencode.media.mit.edu/].
[10] Myron Krueger. *Artificial Reality 2*. Addison Wesley, 1991.
[11] Golan Levin. [http://www.flong.com/].
[12] Golan Levin. Painterly interfaces for audiovisual performance. Master's thesis, Massachusetts Institute of Technology, 2000.
[13] Casey Reas. [http://reas.com/].
[14] Casey Reas. Behavioral kinetic sculpture. Master's thesis, Massachusetts Institute of Technology, 2001.
[15] Casey Reas and Benjamin Fry. Processing. [http://processing.org/].
[16] Casey Reas and Benjamin Fry. *Processing a programming handbook for visual designers and artists*. MIT Press, 2007.
[17] Randi J. Rost. *OpenGL Shading Language*. Addison Wesley, 2006.
[18] Bob Sabiston. [http://www.flatblackfilms.com/].
[19] W.R. Sabiston. Extracting 3d motion from hand drawn animated figures. Master's thesis, Massachusetts Institute of Technology, 1991.
[20] Norbert Weiner. *Cybernetics*. MIT Press, 1943.
[21] Wikipedia. Etienne-jules marey. [http://en.wikipedia.org/wiki/Étienne-Jules_Marey].
[22] Wikipedia. Marcel duchamp. [http://en.wikipedia.org/wiki/Marcel_Duchamp].
[23] Wikipedia. Myron krueger. [http://en.wikipedia.org/wiki/Myron_Krueger].