

**LE CODE**

# Le Code

Audio-Visual duet.

**Yoann Trellu**: concept, visuals and programming.

**Mangrove Kipling**: sound composition and interpretation.

**Le Code is a live-generated digital painting whose soundtrack is composed by Mangrove Kipling.  
Le Code should be perceived as an abstract movie, not as a concert with visuals.**

The painting emerges from artistic and aesthetic choices:

- Only horizontal and vertical should be used in terms of graphic elements.
- Only plain colors should be used.

This is revealed by colored lines and rectangles.

- At times, an alien element comes to disrupt the order appearing as a dramatic event.

*Le Code* is a contemporary digital interpretation, inspired by ideas from the Futurists, the dawn of abstract art in the early twentieth century, Bauhaus...

"I like the idea of a graphic narrative and I believe that emotions and feelings can be communicated through colors, abstract shapes' movements, geometry. (...)

The other part of my work is to build softwares, "digital machines", using the Max/MSP/Jitter software.

I trust that through automated and semi-randomized actions, it is possible to experiment and to create works that are impossible to achieve without the computing tool.

A great part of my job follows the idea that new technologies allow nowadays the machine to go beyond its tool status and to become creator."

Yoann Trellu

**Le Code is a visual and poetic language put to music by Mangrove Kipling.**

The subject is an attempt to give a depiction of Life on a graphic level.

Or the creation of digital sonic paintings that might induce it.

The graphic evolution can be almost undetectable, or very fast and sequenced.

This alludes to memory, the perception of time, and the notion of evolution.

The basis to *Le Code* is a program developed by Yoann Trellu on Max/MSP/Jitter\*.

This software uses audio signals to generate visuals and interact with them.

The interaction sound/image, although it is physical and mandatory, isn't the substance of this project: the software uses the audio signal to function but its aim isn't its visualization. The relation sound / visuals can be clear, but also absolutely impossible to see.

**A "hide and seek" game between the music and the visuals.**

Mangrove Kipling composes the music live during the performance.

While the visual part of the performance is manifested through graphic representation rhizomes, interactions, intersections and so on, the audio part of it openly formulates chaos and progression as tightly interlaced. The free and non-codified use of sounds (in terms of the way they are played), taken from Nature, or generated by musical instruments, electronic objects or circuit-bent items, allows on the one hand the composition to expand free of any academic constraints, and thus representing a freedom of expression only modified by Mangrove Kipling's physical interaction, while on the other hand the surprising and challenging the audience with the existing connections between video and music, as well as the narrowed subjective relation that binds it to the instant. The choice of particularly non-conventionally selected sounds appears as an accurate symbol of the apparent disarray, which in fact only hides a complex behavior.

Visuals are produced live. They are only influenced by two factors<sup>1</sup>:

- An automatic "mechanical" one, the computer.
- A human one, sensitive.

\*:Max/MSP is a sound software that allows the production of sound synthesis, analyses, and recording and can also be used as a MIDI control. It was developed by IRCAM (Institut de Recherche et Coordination Acoustique/Musique founded in 1969 by Pierre Boulez) in the 80's, and is one of the most popular software among musicians, professionals as well as amateurs.

Jitter is a complementary library added to Max (as is MSP) that allows it to work on matrices. Its range is therefore very broad: live image editing (for Vjaying) but also audio synthesis (FFT spectrographies), mathematical matricing or 3D modelling in Open GL.

1: See Appendix 1, "generating visuals"

# Appendix 1

On generating visuals during a *Le Code* performance

Before stepping into the “how”, we feel it is important to specify that it is absolutely not necessary to understand or even get to know the following in order to enjoy a *Le Code* performance.

## Graphic principles

*Le Code* is made up of three elements:

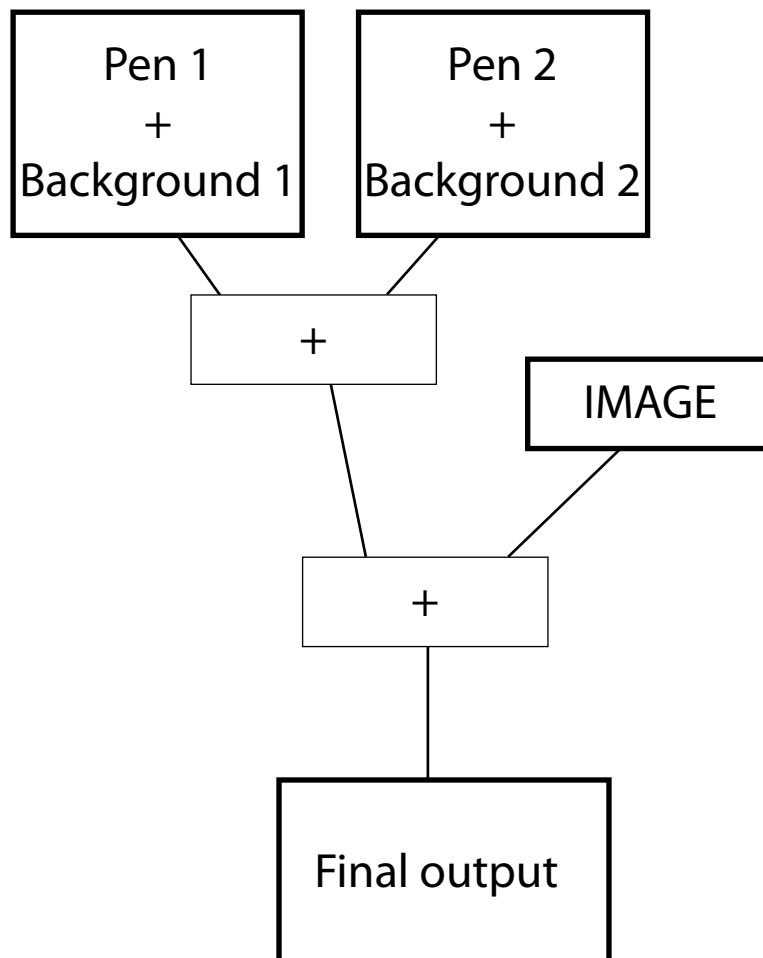
Two markers, each on their own background.

- A marker is a pen of variable size, color and opacity.
- The background can be compared to a paper sheet whose plain color would vary.

Background and markers' characteristics:

- Markers are squares or rectangles, whose dimensions vary on a X and Y scale.
- They draw as soon as they move, never raising off the paper.
- But their color can be completely transparent, or invisible, which is equivalent to raising the pen off the paper.
- By choice, they can only move in a vertical or horizontal direction.
- Color, transparency and size can be changed every time they move.
- Their layouts accumulate until the order to erase is given by the computer or by the user.
- The background color can change every time an erasing phase starts.

The third element is an image that is encrusted on the image of the mixed two first elements. This image is used in order to disrupt the order of the first two elements, and therefore create a dramatic tension. The chosen image is a blurred white circle on a black background.

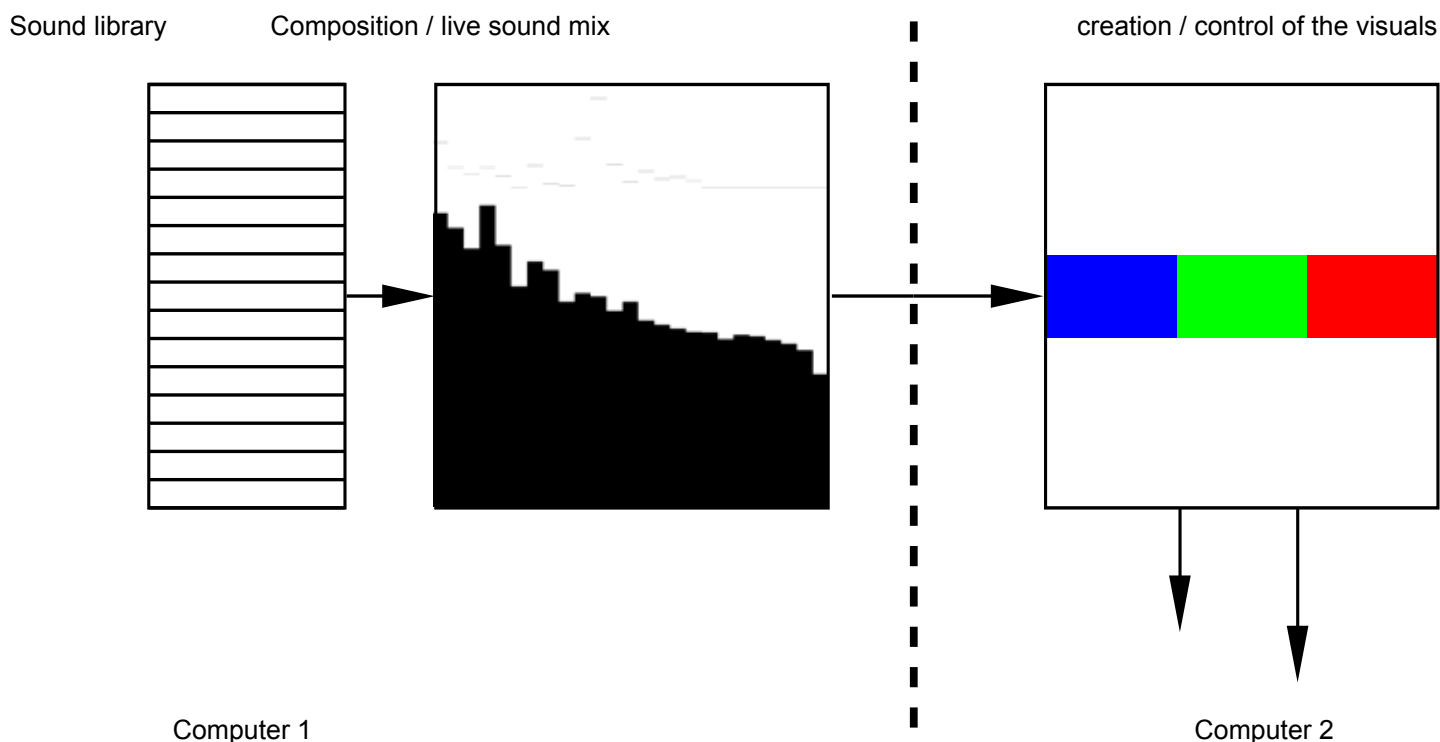


## Generative and interactive principles:

The animation seen during a *Le Code* performance comes from the motion of two pens, these motions, being generated as new coordinates (X & Y), are sent to the pens, ordering them to move. This is triggered by Mangrove Kipling's sounds.

The sound is therefore "turned into" spatial coordinates directing the pens' motion.  
This motion is instantaneous.  
Thus, there are no motions but an accumulation of shapes.

Attacks detected in the audio signal may have an effect of erasing the layouts made by the pens.



The place and shape of these forms is therefore directly controlled by the audio signal, but the software also allows manual adjustments tightly linked to what the sound is or is not allowed to do, and along which value range. These manual actions are possible live thanks to keyboard shortcuts and direct actions with the software.

This is the human part of the software, turning it into a genuine visual instrument.

## Appendix 2

Screen captures / examples of visuals generated by *Le Code*

