

# **HARNESSING THE ENACTIVE KNOWLEDGE OF MUSICIANS TO ALLOW THE REAL-TIME PERFORMANCE OF CORRELATED MUSIC AND COMPUTER GRAPHICS**

Ilias Bergstrom, UCL LottoLab

[www.onar3d.com](http://www.onar3d.com)

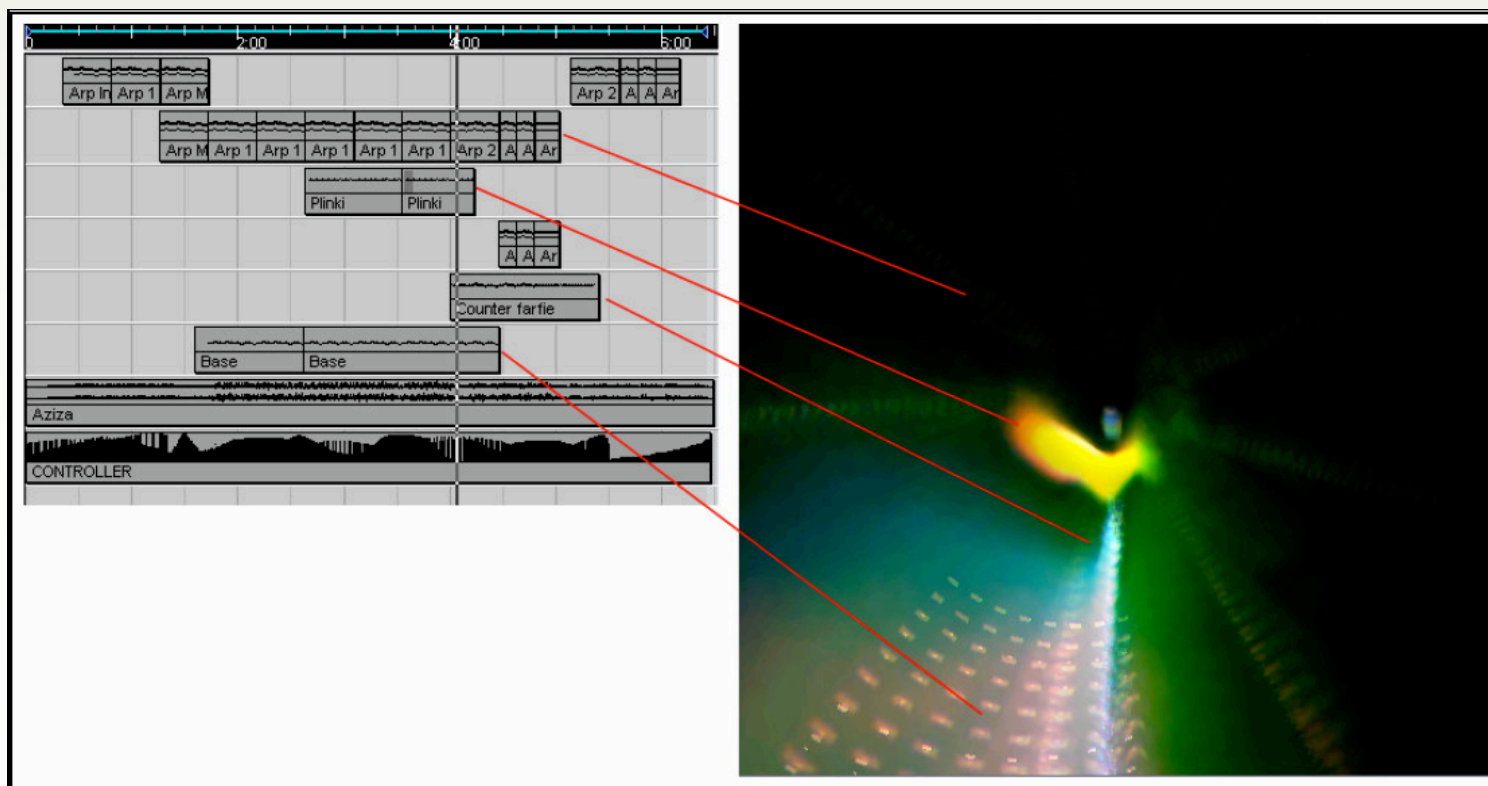
\* Enactive knowledge is that which can only be manifested or learned through physical action: e.g. cycling, painting, or playing a musical instrument.

## I have developed:

- A system comprising of three applications, for real-time performance of interacting synthesized computer graphics and live music, to be used by a single performer, or ideally a group.
- Graphics and music are primarily performed using the same interface, the musical instruments. Additional input from non-musical controllers is possible.
- Hypothesis:
  - The advantages musicians enjoy when performing using instruments, may be transferable to controlling real time synthesized computer graphics.
- There is no hardwired mapping between musical input data and control data for visuals. This is instead set up for each individual performance, and may also be altered during it's course.

# Input

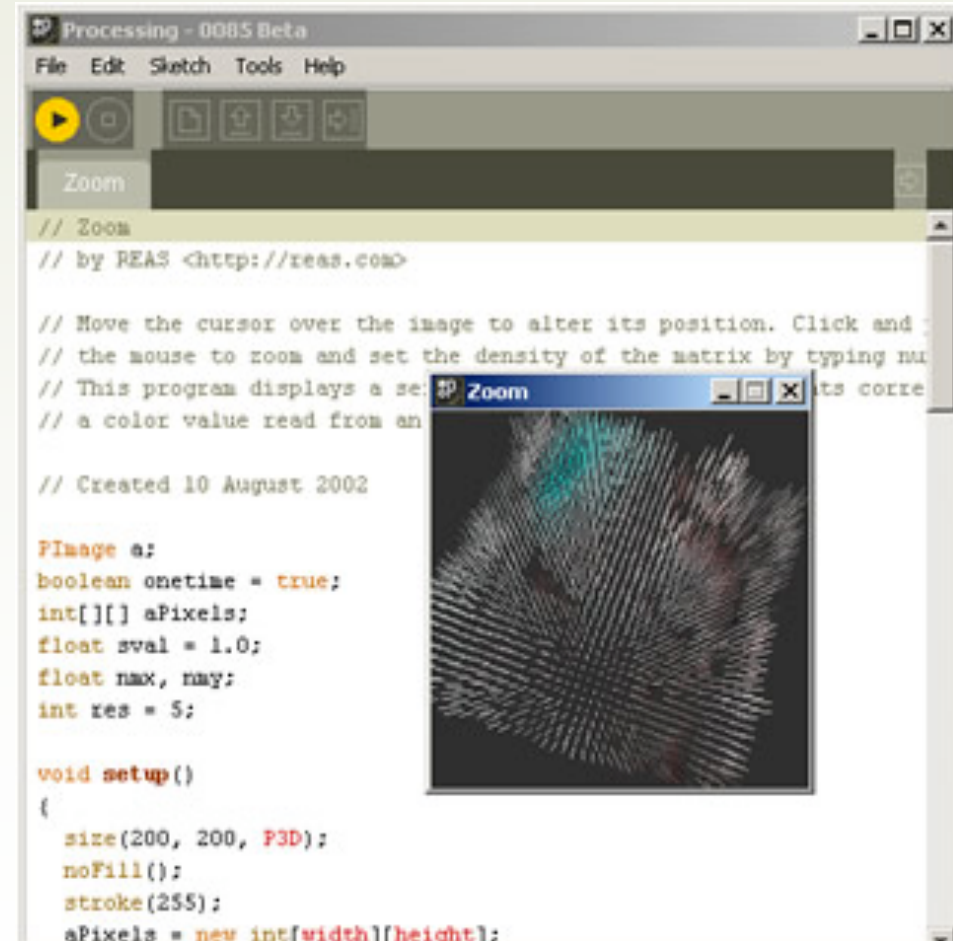
- From all aspects of the musicians performance:
  - scales, notes, chords, tempo, velocity, sound timbre and volume, for each instrument separately.



# Programming for non-programmers: creating visual synthesizers

Processing:

a very simplified language based on Java, which enables artists to program real-time computer graphics.



```
Processing - 0085 Beta
File Edit Sketch Tools Help

Zoom

// Zoom
// by REAS <http://reas.com>

// Move the cursor over the image to alter its position. Click and
// the mouse to zoom and set the density of the matrix by typing nu
// This program displays a se
// a color value read from an

// Created 10 August 2002

PImage a;
boolean onetime = true;
int[][] aPixels;
float sval = 1.0;
float max, min;
int res = 5;

void setup()
{
  size(200, 200, P3D);
  noFill();
  stroke(255);
  aPixels = new int[width][height];
```



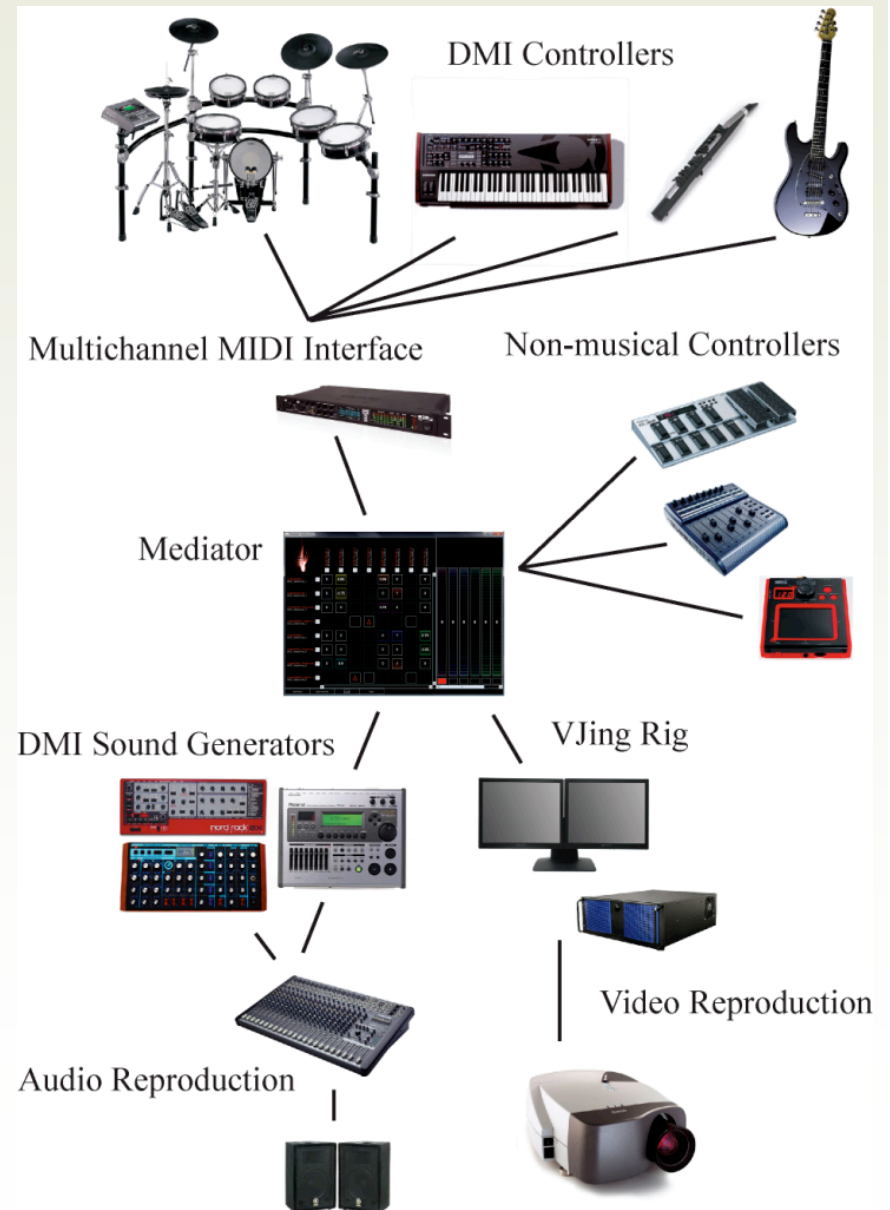
# Steps towards a performance:

1: Creating new Visual Synthesizers, or selecting which to use from a library of existing ones.

2: Setting up a scene, which comprises of a number of visual synthesizers, and a preset for each.

3: Setting up the initial mapping between the input parameter space of the instruments, and the parameter space of the visual synthesizers.

# Signal Flow:



## Applications:

- Input processor
  - Takes multichannel audio input, and accompanying midi. Generates amplitude, pitch and beat data from audio, and transmits it over OSC. MIDI is also translated and transmitted over OSC.
- Mother (free open source)
  - For layering multiple Processing sketches (visual synthesizers), and forwarding OSC control data to these.
- Mediator
  - For dynamically creating and altering mappings between input control data, and control data of visual synthesizers.