



**HAL**  
open science

# Apprehending works of computer music through a representation of the code

Maxence Larrieu

► **To cite this version:**

Maxence Larrieu. Apprehending works of computer music through a representation of the code. International Conference of Students of Systematic Musicology (SysMus 2014), Sep 2014, Londres, United Kingdom. , 10.6084/m9.figshare.4715371.v1 . hal-01371951

**HAL Id: hal-01371951**

**<https://hal.science/hal-01371951>**

Submitted on 3 Mar 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# Apprehending works of computer music through a representation of the code

Maxence Larrieu, Université-Paris-Est, LISAA, UPEM, France

## Summary

My research is based on the code that exists in computer music works. I suggest that the code can be helpful for musical analysis.

Having seen the main difficulties while studying this code, I am introducing a tool that provides a representation of the code during the work

By representing the elements of the code that produce the music, I suggest a new way to apprehend works of computer music.

## Introduction

At the end of the 20th century Jean-Claude Risset highlighted a novel aspect of computer music composition : sound production through calculation (Risset 1999). Pursuing this line of thought, I view composition as a process that defines the different calculations needed to produce music. Since this calculations are implemented in programming languages, I name them 'code'.

## Code is rich

First I notice that the relation between the code and the work is new in music : the same code permit to compose, to actualize and to reconstruct the work. This omnipresence of the code reveals its richness : it contains a lot of information. Thus, as Matteo Meneghini has shown in 2006 with Stria, a study based on this code can be helpful for the analysis.

## Main difficulties

Nevertheless, this approach is not yet widespread in musical analysis. I can understand this by revealing the main difficulties for studying the code :

- Code is hermetic : it is dedicated to the computer, not the human
- Code is plural : several programming languages can be used in a piece
- Codes of works are heterogeneous : many languages are used in computer music
- Languages are not perennial : they are evolving with the technological obsolescence (corollary to Moore's law)

## Theoretically Proposition

I believe that a representation of the code can be helpful for the analysis. Here are the main points of the desired representation :

- Independent to programming languages : it must be applicable in all cases
- "Upper" the programming languages, i.e. in a higher abstraction level
- Readable : all the code do not have to be include but only the salient element
- Personal : the representation is built by the analyst
- Dynamic : the representation must evolve at the same time as the music

Most of these points can be illustrated with the Graphic User Interface (GUI) often present in code of works. Mainly, GUI are used to give a global view of the code so that the user can easily *control* the computation. In a same way, the representation I need must be built upper the code to facilitate the *understanding*.

## Practical proposition

I have developed a software that allows to build this representation. The implementation is done with Processing, a programming language dedicated to graphic rendering. To build the representation, the user has to identify the salient element of the code, construct OSC message, and send it to the software.

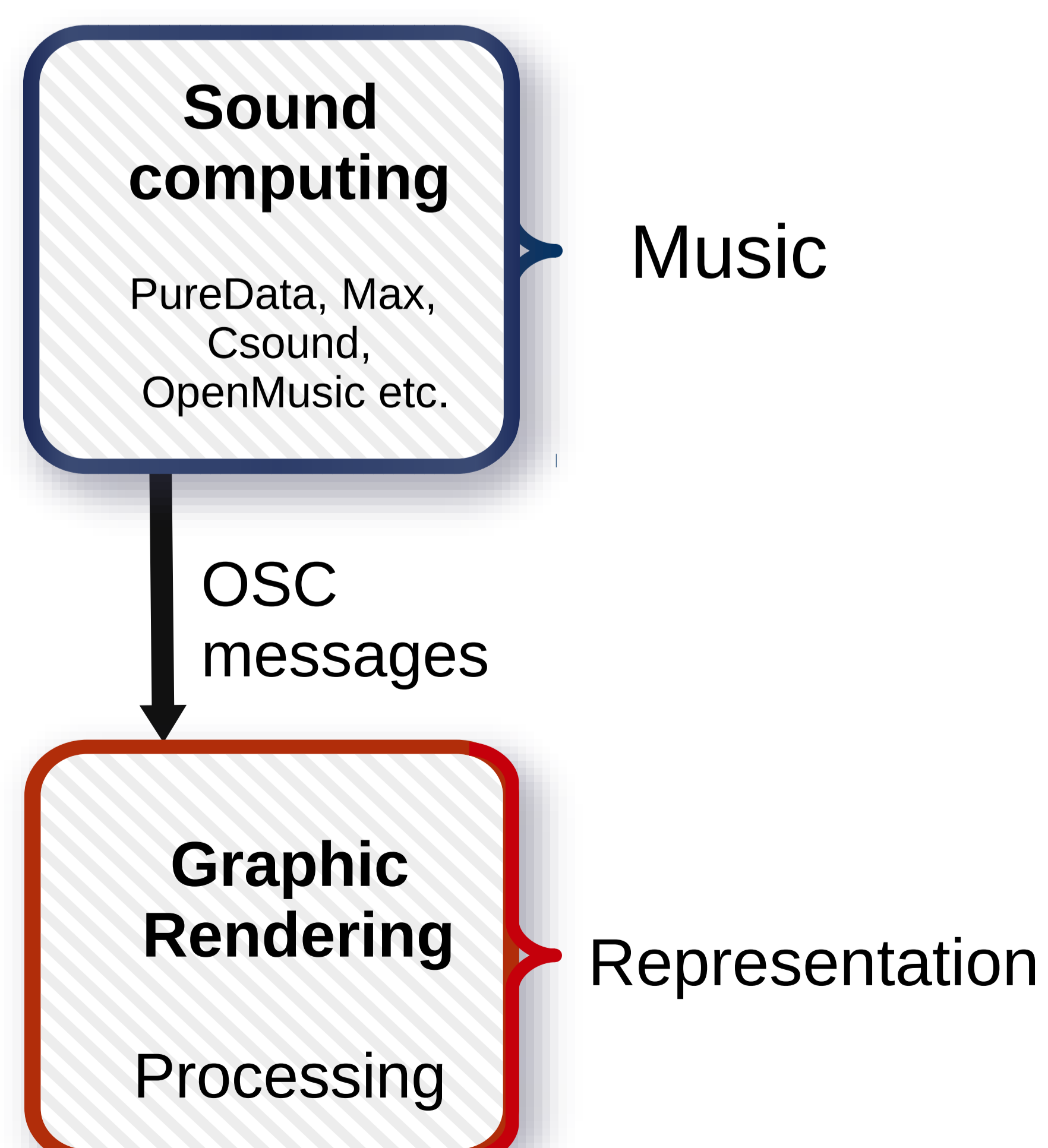


Illustration of the device

## Interactivity

I dedicated these representations to the "interactive works" (i.e. when an exterior phenomena is acquired and linked to the computation). Thus, the representation includes three layer : the inputs, the connections and the computation. I choose to work on interactive pieces because the representation of the code are more meaningful.

## Example

```

vibraStereo
  attack 1 amp 75
frequencyModulation( ampMod 294 )
> lowpass( cutoff 420 )
> tremolo( freq 3 )
  
```

Capture of a representation given by Dodécalite Intérieur, Jean Michel Bossini, 2009, for vibraphone and live electronics

## Conclusion

I introduce a new way to use the code of works in an analytically approach. By identifying and representing the evolution of the most salient element of the code, we would like to permit another way for apprehending electroacoustic and computer music works.

This research also leads to question the relation between the piece of work and the code, this brings to the ontology of computer music works.

**Maxence LARRIEU**  
Université Paris-Est  
Laboratory LISAA

SysMus 2014