

Deep dive in the ossia ecosystem

with libossia and ossia score and friends

Jean-Michaël Celerier

LaBRI, Blue Yeti, GMEA

February 2, 2019

A small video example

What it is

- ▶ Tooling for audio-visual installations, live shows, museum exhibitions, etc etc
- ▶ Free software (**GPL / LGPL v3**)
- ▶ Cross-platform (Linux, macOS, Win, Android, iOS, ARM, PPC...)
- ▶ Qt (GUI) / Modern C++ (libraries).
- ▶ Compatible with a lot of existing environments : Processing, OpenFrameworks, Python, QML, Unity3D...

libossia

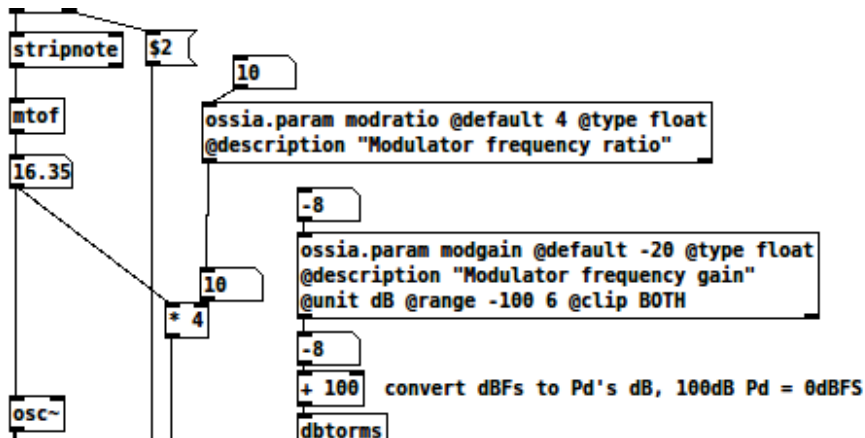
Let's define stuff

libossia

A distributed object model.
Based on the Open Sound Control specification.

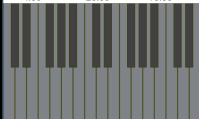
📁 ossia-c	many improvements to the Java API for Processing	13 days ago
📁 ossia-cpp	convert std::string_view to std::string in cpp98	7 days ago
📁 ossia-faust	New portaudio-dynamic-arch.cpp for libfaust + LLVM based dynamic comp...	a year ago
📁 ossia-java	many improvements to the Java API for Processing	13 days ago
📁 ossia-max	[ossia-max] remove per-object timer, use global timer instead	a month ago
📁 ossia-node	minor cmake improvement	5 months ago
📁 ossia-pd	fix pd build (TODO check max build)	2 days ago
📁 ossia-python	fix #461	16 days ago
📁 ossia-qt	fix failing tests	7 days ago
📁 ossia-unity3d	preset fix	8 months ago
📁 ossia	various fixes: windows tests, phidgets protocol	2 days ago

Demonstration



connected to device ws://127.0.0.1:5678

decay

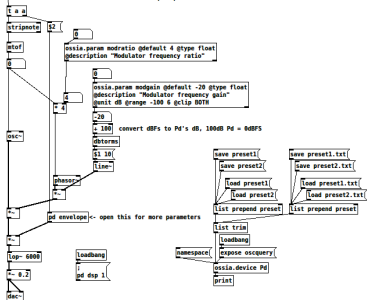


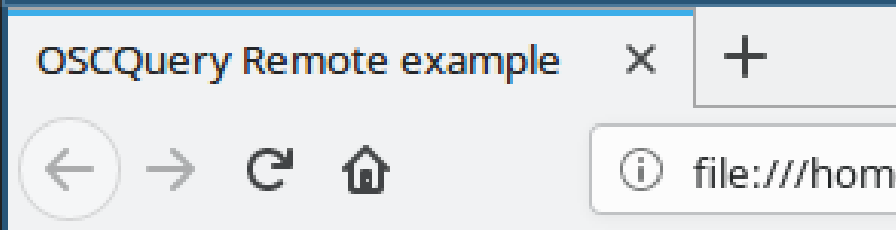
You can use your computer keyboard to play

Fichier Édition Ajouter Rechercher Média Fenêtre Aide

A simple synth to demonstrate communication between Ossia device and client.

You can use - the patch "client-example.pd",
- the "client-example.html" web application
- or the QML application
"client-example.qml" to control this device.





Play



Note

Velocity

Mod. ratio



Mod. gain



Interoperability: welcome OSCQuery

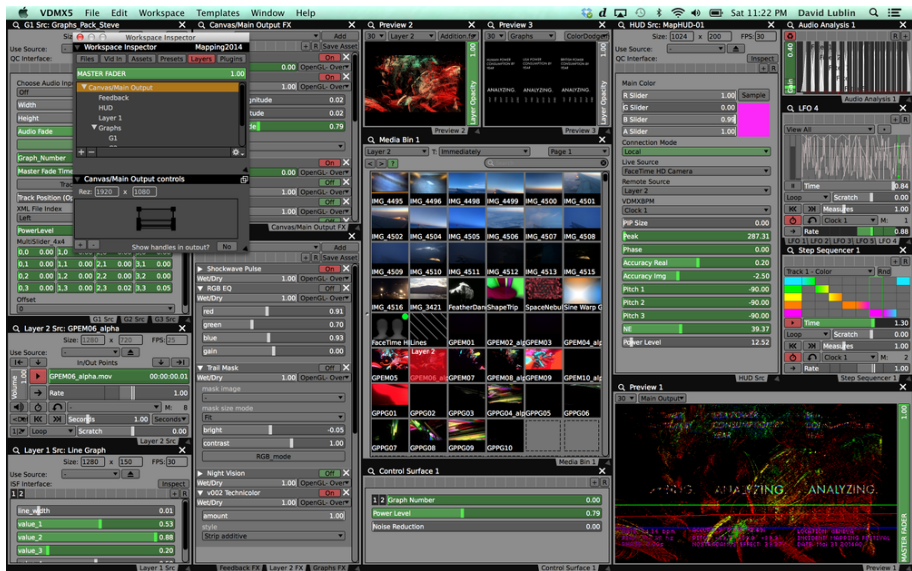
Initiative spanning the industry.

VDMX, Millumin, MadMapper, Vezér, and ossia.

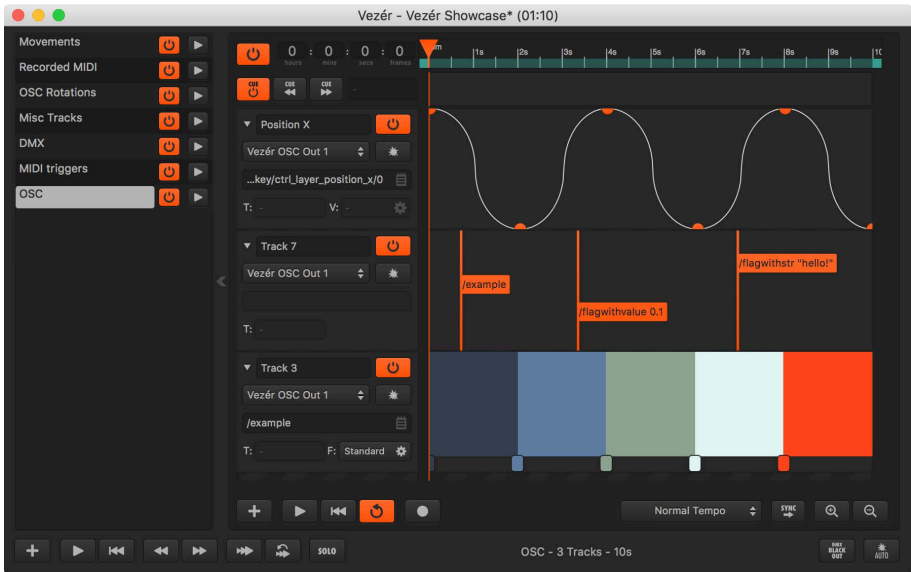
Multiple compatible implementations.

Retrocompatible with existing OSC implementations (liblo, oscpack...).

github.com/Vidvox/OSCQueryProposal



Vezér



Millumin

The screenshot displays the Millumin software interface, which is used for designing and controlling stage lighting. The interface is divided into several sections:

- Top Bar:** Includes a window title "Lights.millumin", an "Optimize" button, and icons for "Output" (monitor) and "Interactions" (keyboard).
- Right Panel:** Contains "Help" and "Contact" buttons, and a list of light parameters with sliders and color pickers:
 - Red 1, Green 1, Blue 1 (each with a color circle and a 0 to 100% slider)
 - Dimmer 2 (with a gear icon and a 0 to 100% slider)
 - Color Temperature 2 (Linear) (with a lightbulb icon and an "RGB Mode" dropdown)
 - Red 2, Green 2 (each with a color circle and a 0 to 100% slider)
- Main Stage View:** A top-down perspective of a stage with rows of orange seats. Three circular light fixtures are highlighted in the center. A red line indicates a path or boundary. A button labeled "Exclude From List" is visible.
- Timeline Section:** Located at the bottom, it includes a "Dashboard" dropdown, a "Timeline 1" tab, and a "+ timeline" button. The timeline itself shows a sequence of events with a red triangle marker at 01:00. Below the timeline, there are two tracks for "ALC4 1" and "ALC4 2". "ALC4 1" contains two dimmer tracks with yellow keyframe curves. "ALC4 2" is highlighted in orange and contains a single yellow keyframe curve.
- Bottom Bar:** Features a "+" button, a "setup" button, and two tabs: "keyframes/segments" and "curves".

The protocol

- ▶ JSON over WebSockets : far from efficient, but works everywhere
- ▶ Supports WebSockets directly (no need to have bridge WS-to-OSC software).
- ▶ “Unit” system : color conversions, position, distance, loudness, frequency...
- ▶ Allows choosing to send data over UDP and TCP, per-OSC address.
- ▶ PubSub-like mechanism - useful when you need only two OSC nodes out of 15000.
- ▶ All implementations support ZeroConf / Bonjour discovery.

JSON Raw Data Headers

Save Copy Collapse All Expand All

FULL_PATH: "/"

▼ CONTENTS:

▼ decay:

FULL_PATH: "/decay"

TYPE: "f"

VALUE: 500

▼ RANGE:

▼ 0:

MIN: 0

MAX: 1

ACCESS: 3

CLIPMODE: "none"

▼ UNIT:

0: "time.ms"

DEFAULT_VALUE: 500

▼ TAGS:

0: "envelope"

REFRESH_RATE: 10

ossia remote

Let's control stuff

Demonstration



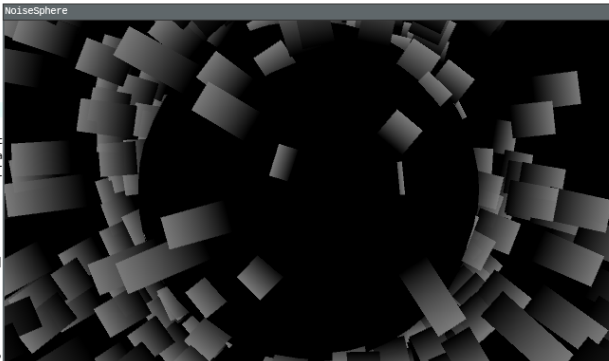
NoiseSphere ▼

```

48     }
49     default:
50         return color(0);
51     }
52 }
53
54
55
56 Protocol p = new OscQueryServer(1267, 5628);
57 Device d = new Device(p, "noise_sphere");
58 Node root = d.getRootNode();
59
60 Parameter c;
61 Parameter p1;
62 Parameter p2;
63 Parameter weight;
64 Parameter ratio;
65
66
67 int cuantos = 500;
68 Pelo[] lista ;
69 float[] z = new float[cuantos];
70 float[] phi = new float[cuantos];
71 float[] largos = new float[cuantos];
72 float rx = 0;
73 float ry = 0;
74
75 void setup() {
76     size(640, 360, P3D);
77     lista = new Pelo[cuantos];
78     for (int i=0; i<cuantos; i++)
79         lista[i] = new Pelo();
80 }
81 noiseDetail(5);
82
83 c = root.create("/color",
84
85 p1 = root.create("/p1", "float");
86 p1.push(200);
87 p1.setDomain(new Domain(0, 300));

```

NoiseSphere



**Address**

▼ noise_sphere

▶ pelos

color

p1

p2

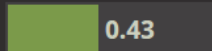
weight

ratio

noise_sphere:/ratio



noise_sphere:/p1



noise_sphere:/weight



noise_sphere:/p2



Pd:/play

ossia score

Let's control stuff (in time)

What i-score is not :

- ▶ PureData (yet)
- ▶ Ableton Live (yet)
- ▶ Bug-free (yet ! 😊)

Does not operate on its own !

- ▶ It's a control center

What i-score is not:

- ▶ PureData (yet)
- ▶ Ableton Live (yet)
- ▶ Bug-free (yet: ☹)

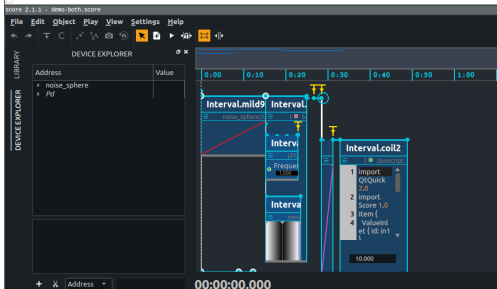
Does not operate on its own!

- ▶ It's a control center

- ▶ Now supports low-latency audio playback (like any other DAW).
- ▶ Useful on its own for musicians.
- ▶ Added cables between processes: DAG-based execution model (like most media software).
- ▶ New semantics are needed to make it work in a timeline:

Authoring interactive media: a logical & temporal approach, Jean-Michaël Celerier, 2018

Demonstration



Faust

- ▶ `faust.grame.fr`
- ▶ DSL for digital audio processing.
- ▶ Generates extremely optimized native code through LLVM.
- ▶ Integrated in score.

Demonstration

File Edit Object Play View Settings Help

DEVICE EXPLORER

Address	Value
processing	[0.0, ...
pelos	14.4282
pl	9.22771
pl2	35
weight	179
ratio	

scenario-processing-noise /

Intervalreef8

Intervalnary11

Interval.hall96

Interval.hera71

pattern

Step sequencer: 1

Envelope: 1

Custom Mapping (Values)

Expression (ExprTK)

$a = 100$

Frequency

Amplitude

Phase

Waveform

Inspector

Step sequencer

Count: 4

Duration (samples): 10000

Min: 50.00

Max: 200.00

00:00:00.000

Basic Ports

ossia videoplayer

(no relevant subtitle to add)

Video player

- ▶ Based on libmpv.
- ▶ Multiple videos exposed over OSCQuery.
- ▶ Easy HAP playback on GL surfaces 🍷

```
3 text files.  
3 unique files.  
0 files ignored.
```

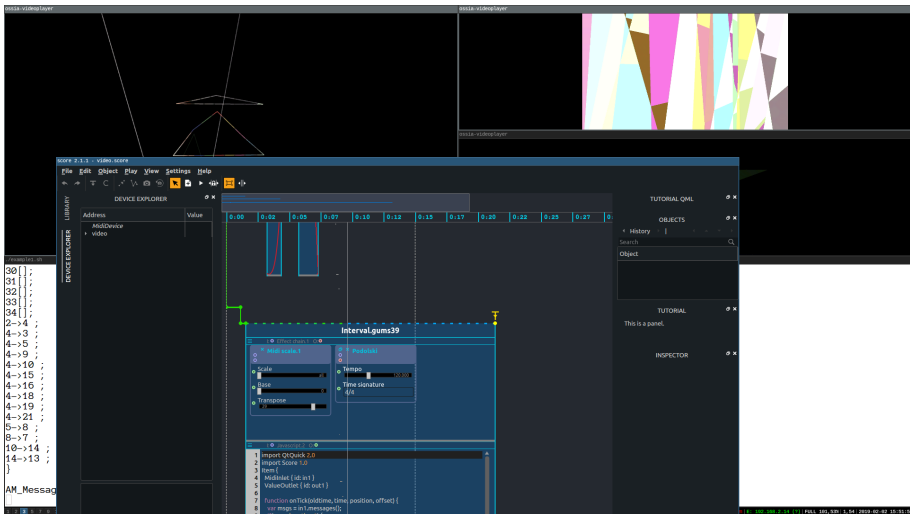
```
github.com/AlDanial/cloc v 1.80  T=0.00 s (731.2 files/s, 82136.1 lines/s)
```

Language	files	blank	comment	code
C++	2	38	10	225
C/C++ Header	1	10	1	53
SUM:	3	48	11	278

HAP

- ▶ Thanks VidVOX (VDMX) again.
- ▶ `github.com/Vidvox/hap`
- ▶ Basic idea: frames compressed with GPU texture compression.

Demonstration



Future works

- ▶ Allow C++ scripting with LLVM (already works fine on Linux, a few more things to tweak on Mac / Windows).
- ▶ Deeper video playback integration.
- ▶ Work-in-progress port to WASM (lacks threads).
- ▶ Fix dem bugs.

Links

- ▶ **ossia score**
`github.com/OSSIA/score/releases`
- ▶ **libossia :**
`github.com/OSSIA/libossia`
- ▶ **ossia remote** (alpha-quality !)
`github.com/OSSIA/remote`
- ▶ **ossia videoplayer** (alpha-quality !)
`github.com/OSSIA/ossia-videoplayer`
- ▶ ... people with OCD, sorry for the naming
- ▶ **Official website :**
`ossia.io`

Thanks ! Questions ?

Credits: 'simple' Beamer theme, Facundo Muñoz; Fira font