# INTERMEDIA ART WITH OSSIA SCORE

*Jean-Michaël Celerier*

SCRIME
Université de Bordeaux, France
`jeanmichael.celerier@gmail.com`

## ABSTRACT

This workshop will introduce newcomers and experts alike to the ossia score software and ossia environment. ossia score is a sequencer for interactive art, able to work with mixed media: audio, video, and most remote-controllable software and hardware through standard protocols such as OSC, MIDI, ArtNet or even raw serial port messages.
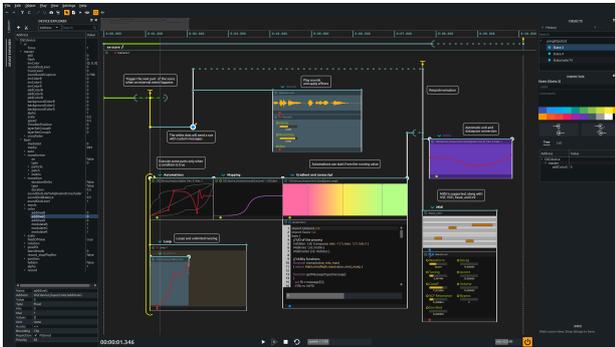
## 1. INTRODUCTION



Figure 1: *ossia score*

## 2. WORKSHOP CURRICULUM

This workshop's goal is to teach the usage of ossia software to the participants.

The first hour will be an introduction to the basic features, by example. The secound hour will task participants in building their own score, through hardware and software made available beforehand.

Participants will be able to use score on its own, or along with software of their choice, such as PureData, or Processing. Examples of usage with various creative coding environments will be provided.

By the end of the workshop, they should have knowledge of the data model of the software, and be able to write their own scores which control remote software and hardware as well as play sounds and apply real-time effect processing on them.

### 2.1. Installing the software

The latest version can be downloaded from `https://ossia.io`. The user library can optionally be fetched from `https://github.com/ossia/score-user-library`.

### 2.2. Playing a sound

To play a sound, it is sufficient to drop it from a file explorer, or the library tab, into the main score. Note that for playback to work, the sound has to be linked in some way to the beginning.

To play, pressing `space` should be enough. If it does not work, please check the audio configuration.

### 2.3. Automating a control

To do that, a Faust process will be added to the score, in order to have a controllable sound generator. One can be chosen from the user library.

Then, creating an automation can be done by:

- By dropping it from the Library (Process) tab.

- Or pressing the + sign on top of the interval.

- Or dropping the port of the Faust parameter you wish to control on the interval.

If either of the two first possibilities was used, the output of the automation must then be connected to the input of the port by dragging and dropping a cable between those ports.

The automation can be edited with various keyboard shortcuts - `Alt`, `Shift` for the curvature, `Ctrl` to use a pen-like tool.

### 2.4. Sending OSC messages

This step is a tad more involved. The context-menu of the device-explorer (available via right-click on the left panel), allows to add devices. After adding a new device using the "Add device" action, the device-protocol can be selected and configured.

Then, it becomes possible to right-click on the newly created device and select "Add child" to add OSC nodes manually.

An OSC address can be dragged to the output port of an automation. When playing, the automation will send its values to the OSC port.

Other processes can be used to send OSC messages: LFO, various mapping processes, the control surface process...

### 2.5. Receiving OSC messages

If OSC is used, it is possible to right-click on the device in the device explorer to perform an OSC learn. This will add all the OSC addresses matching received messages.

## 2.6. Handling interaction

This part will introduce the interactive primitives of the ossia score system. Triggers allow to trigger part of a score only when an external event happens. Conditions allow to execute or not some parts of the score depending on the state of the outside world.

To add a trigger, it is necessary to select a state (a dot) and press the "Add trigger" button. A node from the device explorer (the left panel) can then be dragged on the expression editor (the right panel). Expressions can also be added manually.

The procedure for conditions is almost the same (the "Add condition" button must be pressed instead).

## 2.7. OSCQuery

OSCQuery allows to automatically expose OSC nodes, along with useful and relevant metadata. A quick overview of the protocol will be given, along with example applications.

## 2.8. Applying sound effects

In this part the dataflow features of ossia score will be showcased, by adding effects on a sound file.

To do so, a sound file must be dropped on the timeline. Then, it becomes possible to add an effect process (VST, LV2, Faust...) and connect a cable from the sound file to the effect process.

## 2.9. Video playback and control

In this part the video features of ossia score will be showcased.

The first step is to create a viewport in which ossia score will render. This is done by adding a "Window"device in the device explorer.

Then, the first step is to play a video: just like for sound files, a video file can be dropped on the timeline. Its output port must be set to the newly created Window device. Playback can then start.

Video filters using "Interactive Shader Format" shaders can be applied, just like audio filters.

## 3. ACKNOWLEDGMENTS