

M^2

Alberto Zin*

March 22, 2008

1 Introduction

\mathfrak{M}^2 (which stands for Mondrian 2) is a noise generation and visualization machine for pd. It's the follow on of my previous Mondrian audio toy¹ with some additional features (harmonizer, load/save, direct keyboard control, spectrum viewer).

2 Dependencies and Install

has three main dependencies:

- *creb* library by T. Schouten
- *cyclone* library by K. Czaja
- *grid* from unauthorized library by Y. Degoyon. Grid version > 0.7 is required² (for grid color handling).

plus the usual *expr*. Extended Pure Data distributions by H.C. Steiner³ contains all these three libraries. On File - Startup, add the following keys: “creb”, “cyclone”, while for “grid” add the path to the “unauthorized” library, (File - Path: usually $< your_pd_home > /extra/unauthorized$, at least this is how my settings are).

The core of M^2 is the cross frequency modulation object `xfm~` from the creb lib. The visualization is done by `scope~` from the cyclone library and a spectrum viewer. Main controls are handled in the gui by few sliders and the `grid` external. M^2 was tested in Linux and WinXp using the extended pd version 0.39 (final release). With PD extended version ≥ 0.40 the GUI is almost completely messed up due to the new patch and cords coloring scheme adopted. Nothing can be done at the moment, but it looks quite clean in the Vanilla version of PD or in the final 0.39 extended version (see Figure 1). The CPU load is minimal, nearly 4-5% on my Athlon 2400+.

*Alberto.Zin@gmail.com (or Alberto.Zin@poste.it)

¹<http://alberto.zin.googlepages.com/puredata>

²It's included in the last PD extended versions since, more or less, 1.5 years..

³<http://at.or.at/hans/pd/installers.html>

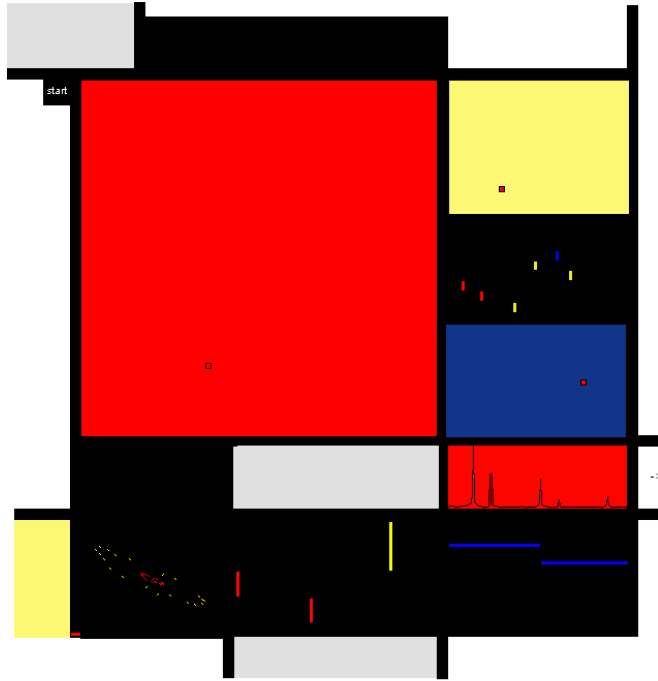


Figure 1: M2 Interface

3 Use

First of all press the black Start button in Figure 1, then move the yellow Volume Slider to the right to rise up the volume. Now you should see something like a growing point on the scope in the lower right part of the patch. Press the "get sound from slot" buttons to hear the pre-loaded sounds. Then tweak with the Grid control #1 , #2, #3 and with the `xfm~` in Figure 1 and hear the results. When a sound is particularly "harmonic", try to tweak the two red "harmonic sliders" in the bottom of the patch (you can do it using the keyboard, see 3).

4 Load/Store

If a particular sound is nice the user can store it in two ways: or in table (4 positions are available) - the sound is lost after patch closure - or permanently in a file. The first option is activated using the "Save sound on slot" buttons (four slots are available) and the four positions can be recalled by "Load Sound from Slot". In case the sound is stored permanently (using button "Write Sound to file") the user is prompt for a name. File name can be any, in the original directory you find sound1-9.dat as examples. Note that loading and writing to file use slot 1 for storing, so everything inside there will be overwritten.

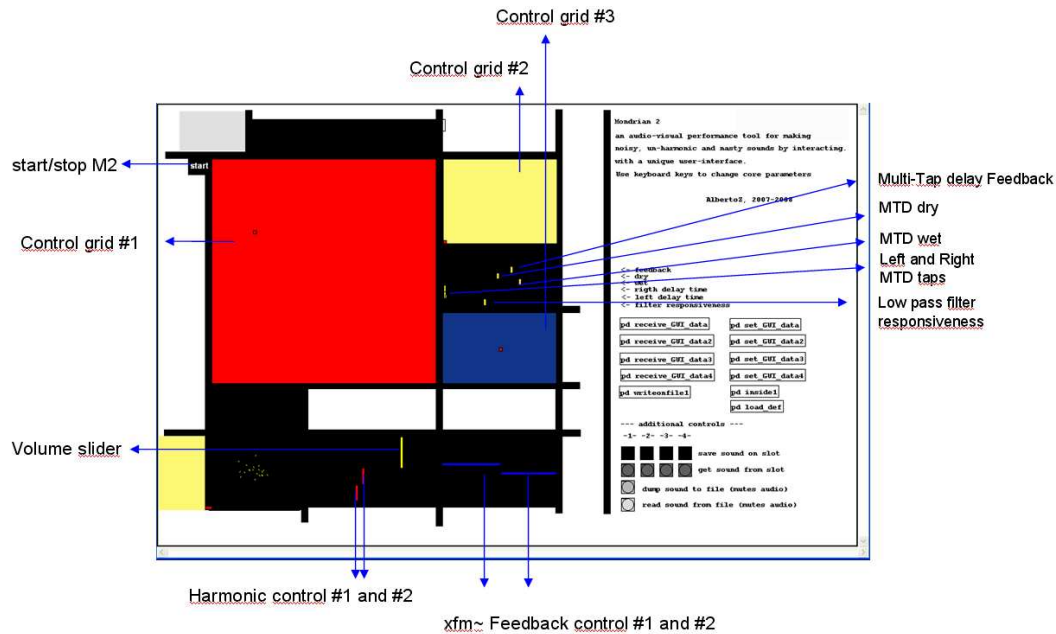


Figure 2: M2 controls

5 Core

M2 core is the `xfm~` external from `creb` library. It is composed by two `xfm~` instances one of which is mixed with a delayed output of itself. You can see it in the hidden `pd` subpatch called `subpatch` (did you find it, didn't you ? :-). The basic scope controls are in the subpatch called `scope_ctr`.

6 Warranty

No warranty at all. The patch is provided “as-is” without any express or implied warranty. In no event shall the author be held liable for any damages arising for use of this patch.

7 License

This patch uses a lot of object grabbed from other people works. Were possible I put the author on the patches. M2 is distributed under the conditions of the GNU Public Licence v.2.

