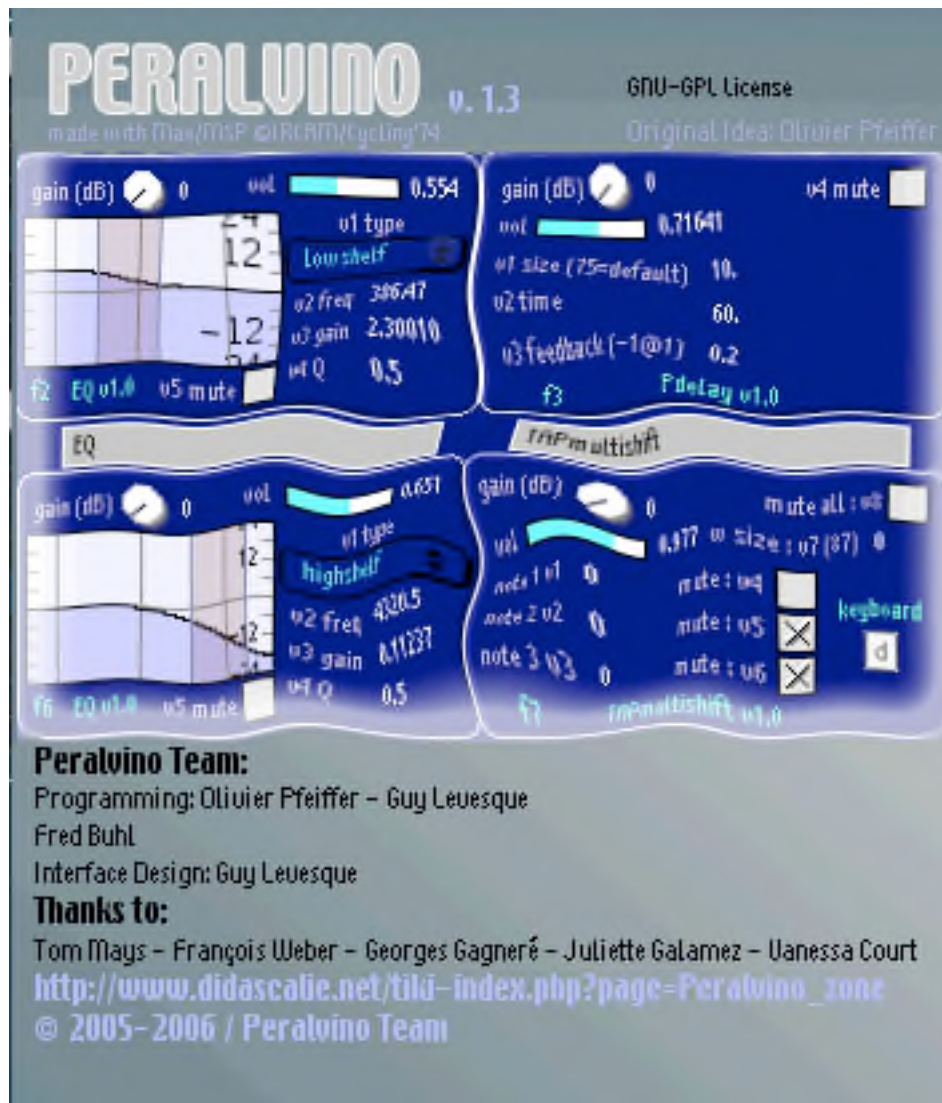




# USER MANUAL





# CHAPTER 1

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

## Preamble

« Peralvino » was developed by Olivier PFEIFFER for Escale #7 (workshop of Didascalie community) in April 2005 at Scène Nationale – La Filature - Mulhouse (France).

This software is a solution developed by Theatre expert who wish to use a tool more suitable for theatre production. After lots of discussions, the software have benefit from the help of Vanessa, Pedro, Jonathan, Guy, Gideon, Nicolas, Jash, Georges and the Didascalie community and François, Tom for sharing some patches under Max/MSP.

Peralvino is a contraction of theatre author's name : Georges Perec and Italo Calvino.

Peralvino was developed in 2 steps.

- Version 1.2 hardcode is a consequence of a meeting with Tom Mays ([www.tommays.net](http://www.tommays.net)), with who we have a close collaboration in production design such as "la Pluralité" (cie Incidents Mémorables). We implement cue list functionalities for an entire production (*janvier 2006*)
- Version 1.3 was designed to be the more accessible, ergonomic and usable in theatre production. Guy design the interface, new functionalities and project development. Olivier, Fred, and Juliette, gave their opinion on phase 1.3. This version is a collective work, done by theatre team engaged in creation production. (*mars 2006*)

We hope that this software will make some echoes in the field of tool in the performing art community.

## About this manual

This manual have seven chapters :

This chapter is an introduction

Chapter 2 explain how to install and spread some important information to understand what is Peralvino.



Chapter 3 is a Step by Step manual which cover all windows and functionalities. Please read this chapter first.

Chapter 4 give references on User Preferences (Audio, Shortcuts, MIDI and Folder) and all available menu.

Chapter 5 focus on input, effects , spatial module and Console window. This Chapter explain how to use script message to control each audio process and time management.

Chapter 6 present Cue list and Conduite functionalities

Chapter 7 give you all specifications about programming and building your own module in Max/MSP. (under construction)

Help us to write a better documentation: if you find some mistakes or a non very useful information, please feel free to send to us your comments.

In order to improve Peralvino, we invite you to send to the Peralvino-Forum, all questions, comments, bugs: [link to the Forum](#)

### **How to report a bug :**

Please follow all message with Peralvino version, your Hardware list (PPC, Intel, RAM, Audio Hardware...). Explain step by step. If we can reproduce bug, we can correct it !!

## What's news in Peralvino1.3

News listing since 1.2 version

### **General**

- Completely new Interface Design
- Added Audio Console
- Preset System implemented

### **Output Console**

- Add 4 parametric Eqs by track
- Modularisation of output number
- Delay parameter on output track

### **Main Console**

- Upgrade to 8 adc (8 direct input)
- A 32x32 Matrix fully usable
- Mute on all tracks

### **Bloc/Slot/Modules**

- 2 new blocs Input: NoiseGen? & Harmonizer
- 1 new bloc Fx : Spectro



# CHAPTER 2

## Description / Installation

### Installation - version 1.3beta-Build

- install the Techno typography (just double-click for Mac os X)
- Drag and drop the Peralvino Preferences Folder into:  
HD:/user/Library/Preferences/
- Move Peralvino1.3beta folder into Applications)
- To start Peralvino, open Peralvino1.3beta Folder, and launch Peralvino1.3.1beta2-build (you can add an icon to your Dock)

Please note that this version is still a beta, functionalities of cue list have been limited to 13 presets.

Under License GNU-GPL © 2006 Olivier Pfeiffer, Guy Levesque  
Please read license and credits text.

### Organisation of Files/Folder With Peralvino

- file PERALVINO1.3.1betax-build (x is the beta number)
- Folder PERALVINOsubfolder  
This folder contains all audio modules and sound folder. It is very important that all your sound files are in the ZSONS folder.
- Folder events  
All cue list file are stored in this folder (never change preset-xxxx and init\_all filename)
- Folder colls  
All preset system instruction are stored in this folder (never change, displace !!)

## Engine / Architecture

Peralvino is a software develop under Max MSP 4.5 (macOSX) for designing and managing real time sound processing in the Performing Art production. It offers lots of possibilities in manipulation, signal processing and spatialisation.

This software allow you to preset, to modify and to reuse all scripted data in a cue list manager (conduite window)



Peralvino allow you to use : 8 adc input (microphone or line input), of 8 Input modules (player, generative sound module...), 8 Effect modules and 8 outputs with your audio hardware at the same time.

Peralvino is an engine. It means that beside each interface object (slider, button...) coexist an address. And you can access by scripting to these addresses. So, writing some command line to the Peralvino engine, you can synchronize effects, recall preset. All addresses and variables form a protocol. This one allow you to work very fast, to edit, to modify and to test some sonic environments during Rehearsal and Design. We hope that this system will fit to the needs of more and more Designer and Sound Operators in sound domain.

## Audio File FOLDER

Sound File must be placed into the ZSONS Folder to be recognize by the Peralvino application.

NOTES: do not name your sound file with space, special characters, accent...

Not good: le son des tarrés.aif

perfect : le\_son\_des\_tarres.aif

## About CPU !

### Audio Set-up

Depending on your computer, set-up your Audio preferences, I/O vector size and Signal vector size.

When you use microphone on stage, reduce to 128 your I/O vector size to minimize the latency of the system.

When you use only sound file player, you can set-up a 512 I/O vector size.

Please, refer to the [MSP45Tutorials&Topics.pdf \(page 8 and 9\)](#) about vector size set-up.

### Output Console set-up

In the Console OUT window, set-up the correct number of output tracks needed.

### Main Console

Mute all unused tracks (adc, input et fx)

Initialize all configuration before starting a new set-up (Menu Conduite->Reset all)

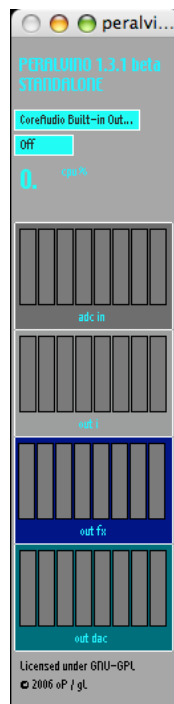


## CHAPTER 3

# SPEP by STEP Presentation

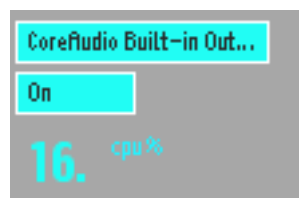
### Open Peralvino

This the main window of Peralvino at start-up. You see Audio Hardware menu and switch and Meters. Here you will control all metering and routing signals.



### Start Audio

Choose your Audio Hardware, and Start Audio. You can set-up all Audio parameters in DSP preferences window.



### Menu Bar

Use it to visualize all functionalities or windows. See Chapter 4 for more information.

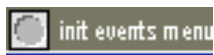
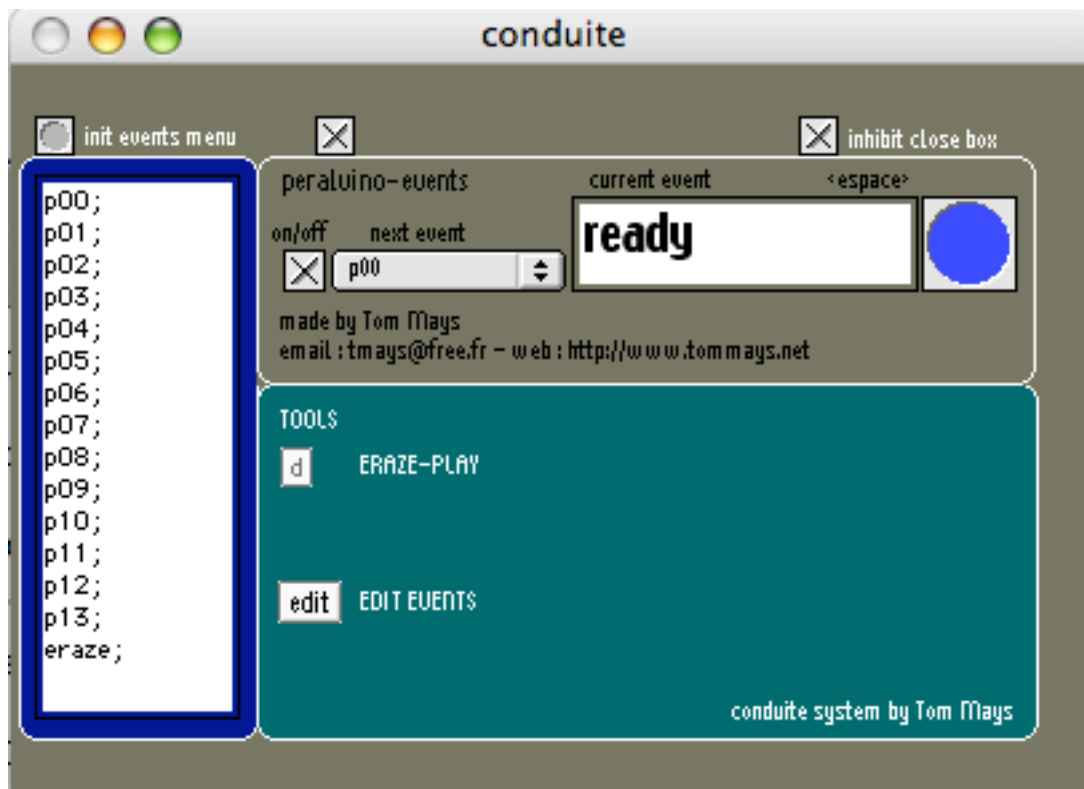
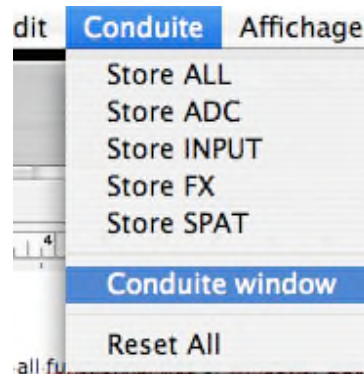




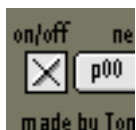
## Conduite (means cue list)

First:

Open the Conduite window (q (minuscule) or Menu Bar/Conduite/Conduite window)



Click on the "bang" init events menu (conduite upper-left window) : ready appears in the menu.



Enable the conduite module by clicking 2 times on the toggle on/off, you should see p00 in the menu. (on=x). p00 is the first preset of our demo, now it's ready to trigger it.

Press 1 time space bar (GO button). You begin to send all data in p00 to the engine.



NOTES: to keep always the status Max window (pomme+m) in order to see what is processing when you trigger cues.

p00: play 1.00000 means that Peralvino is loading, processing data / p00: done means that Peralvino have complete all process and wait for you to trigger the next cue.

Now, you can trigger space bar for hearing the demo and watch some cool functionalities.

CUES:

p00: init 1

p01: Play+fade in of SFmono i1 module

p02: send to FX1 (delay) and automatic fade out of FX after 3 seconds

p03: Moving EQ frequency (From 20Khz to 300Hz in 9s)

p04: Fade out of sfmono

p05: Initialization

p06: init 2

p07: play of sfmono and 5s recording into a GrooveRecPlay<sup>2</sup> i2 module

p08: crossfade between sfmono and groove module (it play 2 time faster than the original (speed))

p09: Fade out of grooveplay

p10: Initialization

p11: init 3 (4 sfmono + 4 spat8)

p12: play of 4 sfmono et start Spatialisation (during 20s)

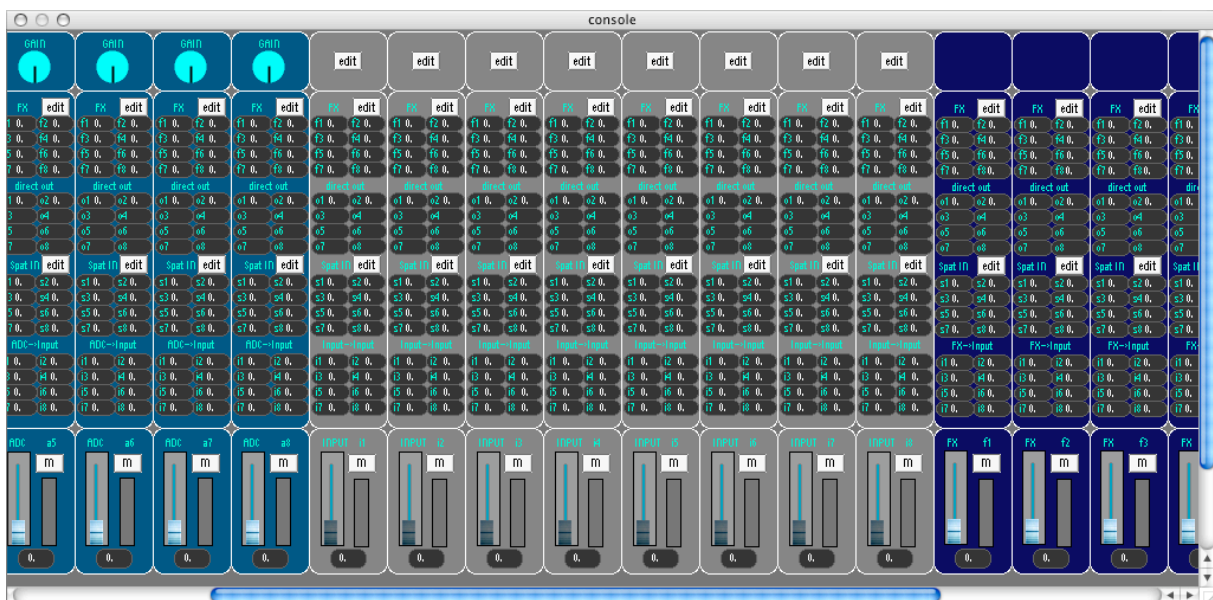
p13: fade out and stop

eraze: Initialization

What do you think about this Demo ? Triggering window to adapt a workspace is a good stuff, isn't it ? let's continue your travel to Peralvino engine...

For more information on Script and Conduite : See Chapter 6

## MAIN Console







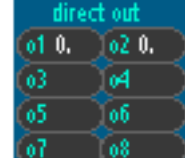

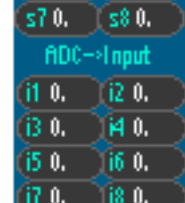
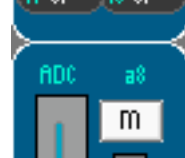




## Description

The Main Console is a Matrix. You can route all audio signal from input modules, adc inputs, send to fx modules, spatial modules and audio direct outputs of your audio Hardware. Some input modules have audio input too (GrooveRecplay), so you can route signal from FX, ADC, INPUT, SPAT to such a module (Buffer modules).


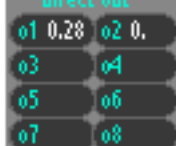
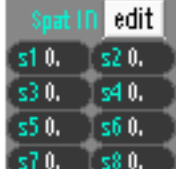

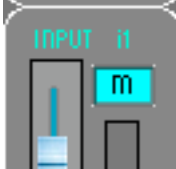
You can use 8 ADC, 8 Input et 8 FX tracks. Spatial modules are mapped to direct outputs tracks.

## Track ADC (a)

	GAIN: gain d'entrée
	EDIT: Visualisation de la page FX
	FX SEND: niveau d'envoi dans Les effets
	DIRECT OUT: niveau d'envoi dans Les sorties (voir Console Out)
	EDIT: Visualisation de la page SPAT
	SPAT IN: niveau d'envoi dans Les spat
	ADC->INPUT: niveau d'envoi dans Les blocs Input (pour Les Grooverecplay...)
	Nom de la tranche (a1->a8) Fonction MUTE Fader Volume Meter Niveau du Fader Volume



## Track INPUT (i)

	EDIT: Visualisation de la page Input
	EDIT: Visualisation de la page FX
	FX SEND: niveau d'envoi dans Les effets
	DIRECT OUT: niveau d'envoi dans Les sorties (voir Console Out)
	EDIT: Visualisation de la page SPAT
	SPAT IN: niveau d'envoi dans Les Spat
	ADC->INPUT: niveau d'envoi dans Les blocs Input (pour Les Groovereplay...)
	Nom de la tranche (i1->i8)
	Fonction MUTE
	Fader Volume
	Meter
	Niveau du Fader Volume



## Track FX (f)

	<p>EDIT: Visualisation de la page FX</p> <p>FX SEND: niveau d'envoi dans Les effets</p> <p>DIRECT OUT: niveau d'envoi dans Les sorties (voir Console Out)</p> <p>EDIT: Visualisation de la page SPAT</p> <p>SPAT IN: niveau d'envoi dans Les Spat</p> <p>FX-&gt;INPUT: niveau d'envoi dans Les blocs FX (pour Les Groovereplay...)</p> <p>Nom de la tranche (f1-f8)</p> <p>Fonction MUTE</p> <p>Fader Volume</p> <p>Meter</p> <p>Niveau du Fader Volume</p>
--	---



# Input PAGE

## Input Page setup

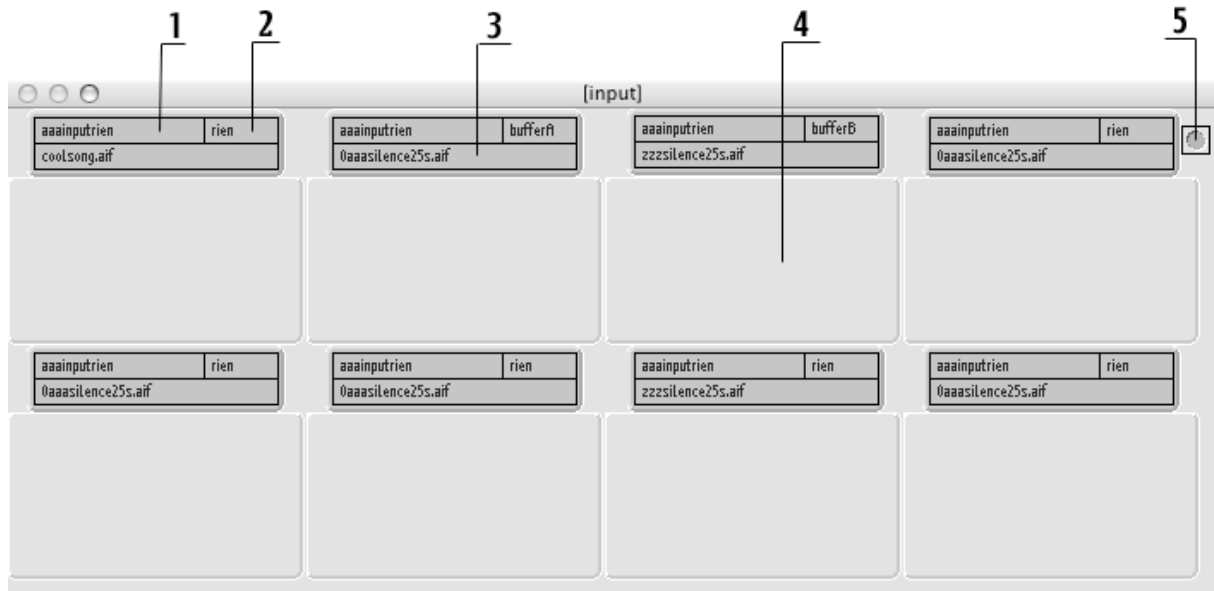
Module selection (1)

Buffer selection (2), Always choose 'rien' for non-buffers modules

Sound file selection (3) which are in the ZSONS Folder. Some generative modules do not need a sound file.

Slots (4)

Reset the page (5)



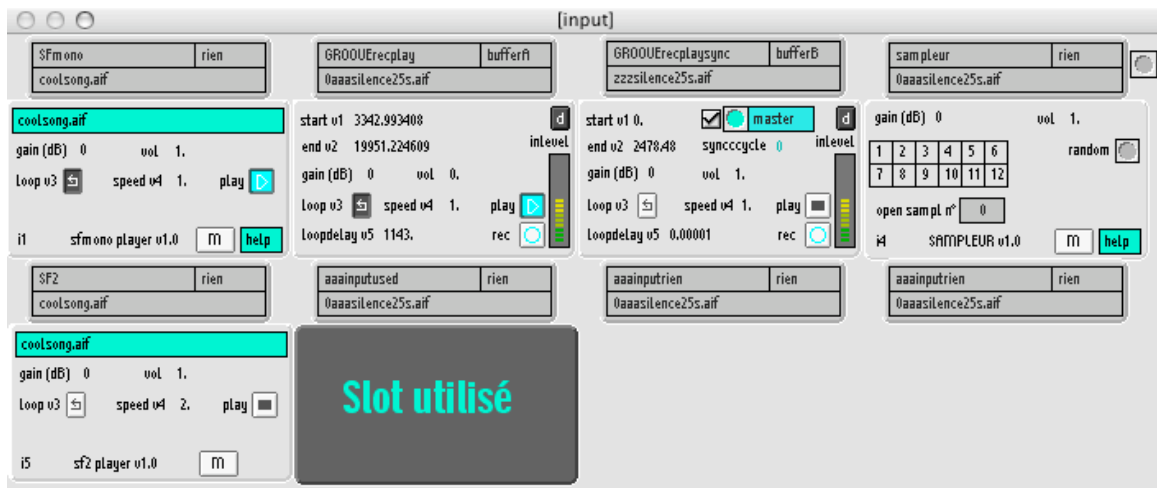
## Specifications

INPUT Page has 8 slots ready to be filled by input modules.

Each Slot is **mono**. Stereo or multi-outputs modules use 2 or more slots they have.

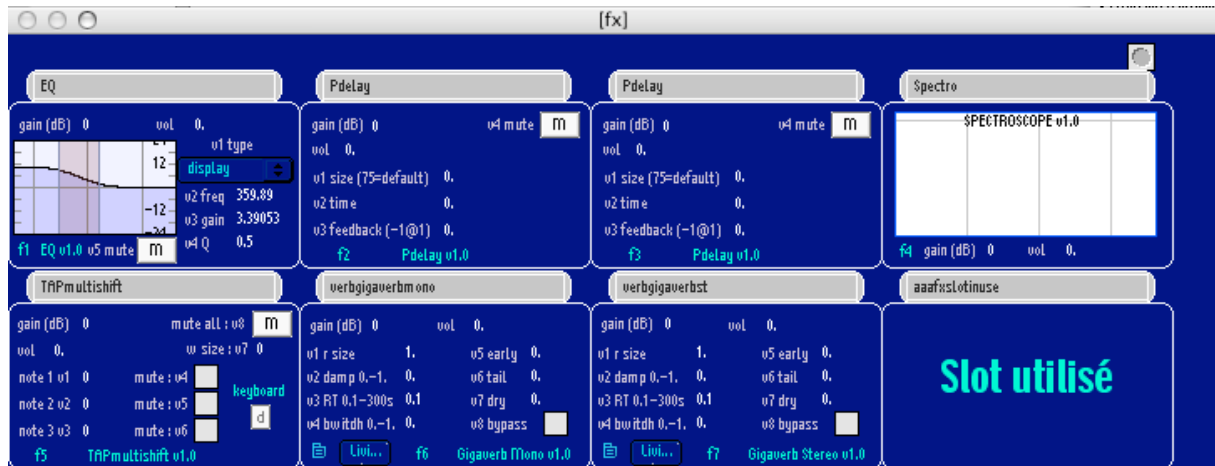
INPUT modules allow you :

Receive a signal, Generate a signal, play a sound file, record a sound.





## FX PAGE



### Description

FX Page has 8 slots ready to be filled by FX modules.

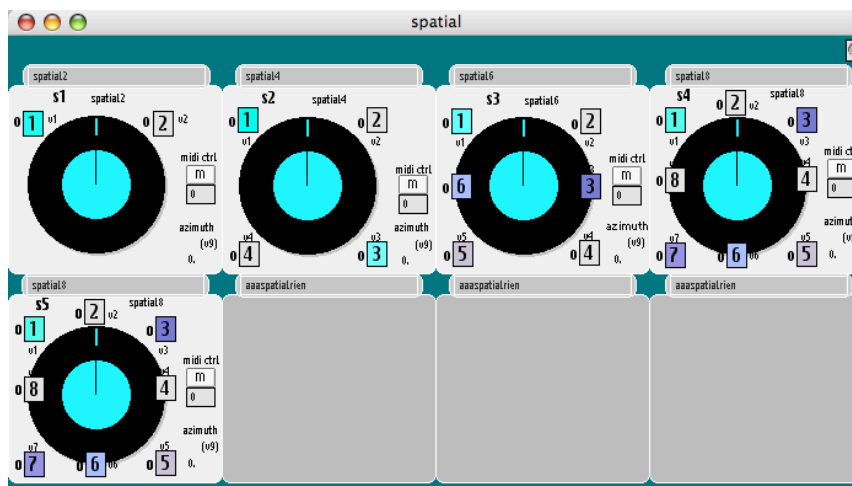
### Pre/Post ?

All FX are Post-fader.

### Insert

You can use FX modules as Insert, look at the main console to do this.

## SPATIAL PAGE



### Description

SPATIAL Page allow you to use 8 slots ready to be filled with spatialisation modules.

Different type of spatialisator are available:  
Spatial 2, 4, 6 or 8 speakers.

All output are mapped to the Outputs Console.



You can set-up the routing directly on the Spatial modules.

## Credits

Copyright 1998-2003 by Ville Pulkki. All rights reserved.

Copyright 2002-2003 by Olaf Matthes. All rights reserved.

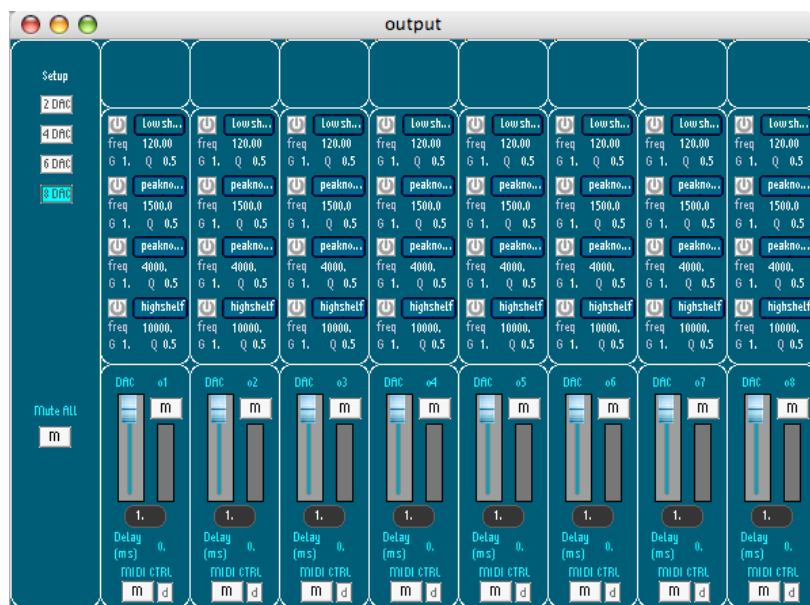
Original VBAP written by Ville Pulkki

Helsinki University of Technology Laboratory of acoustics and audio signal processing  
and UC Berkeley Center of new music and audio technologies

RVABP (reverberated VBAP) written by Olaf Matthes.

"Vector Base Amplitude Panning" and its developer Ville Pulkki

## Output CONSOLE



## Description

Output Console allow you to manage levels, delay (ms), 4 parametric EQs and mute button.

Choose your configuration of output:

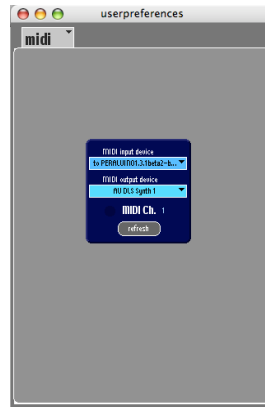
2, 4, 6 or 8 outputs.

Mute all button.



## CHAPTER 4

### MIDI Preferences

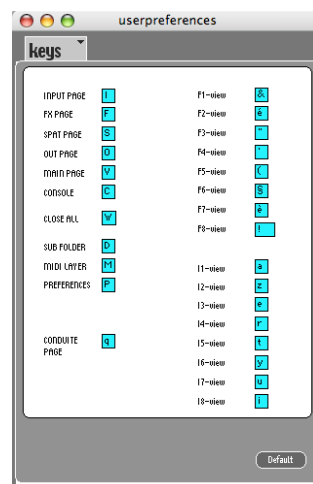


#### Description

MIDI preferences allow you choose Midi In and Out devices. Refresh button is useful to find an new device without restarting Peralvino.

To assign MIDI controller, see the section on page 39.

## Create User Shortcut Keys



#### Precautions

Only keys following are available for shortcuts :

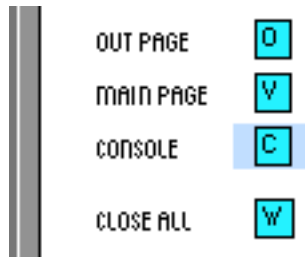
- From **a** to **z** (minuscule)
- From **A** to **Z** (majuscule, shift+letter or caps lock)
- From **&** to **à** (numeric symbol)



## To Edit shortcuts

Open preferences User Keys (shift+p or menu Preference User Keys)

Click on the letter – a blue rectangle appears



Press a key – the letter is now replacing the old one.

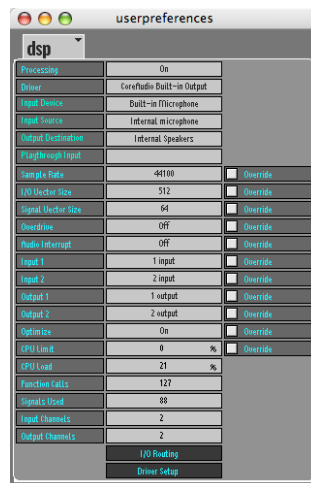


## Default

Back to the default shortcut Key preferences preset

When starting Peralvino, the default preset is recalled. You can not store a Key shortcut preset.

# DSP Preferences



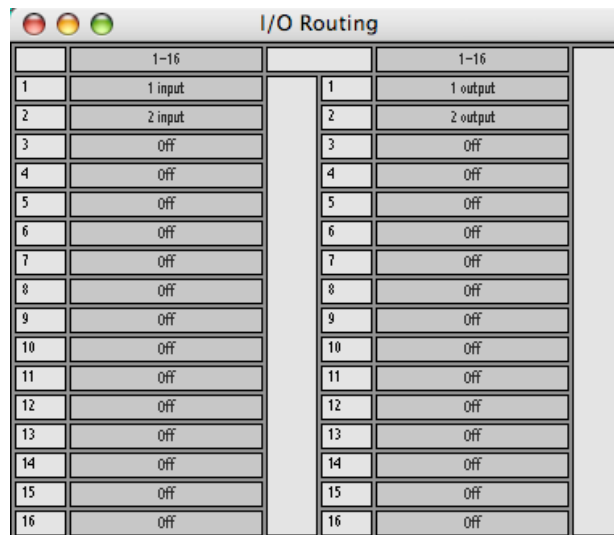
## Description

You can set-up all audio parameters of your system and route all input/output from and thru your Audio Hardware.

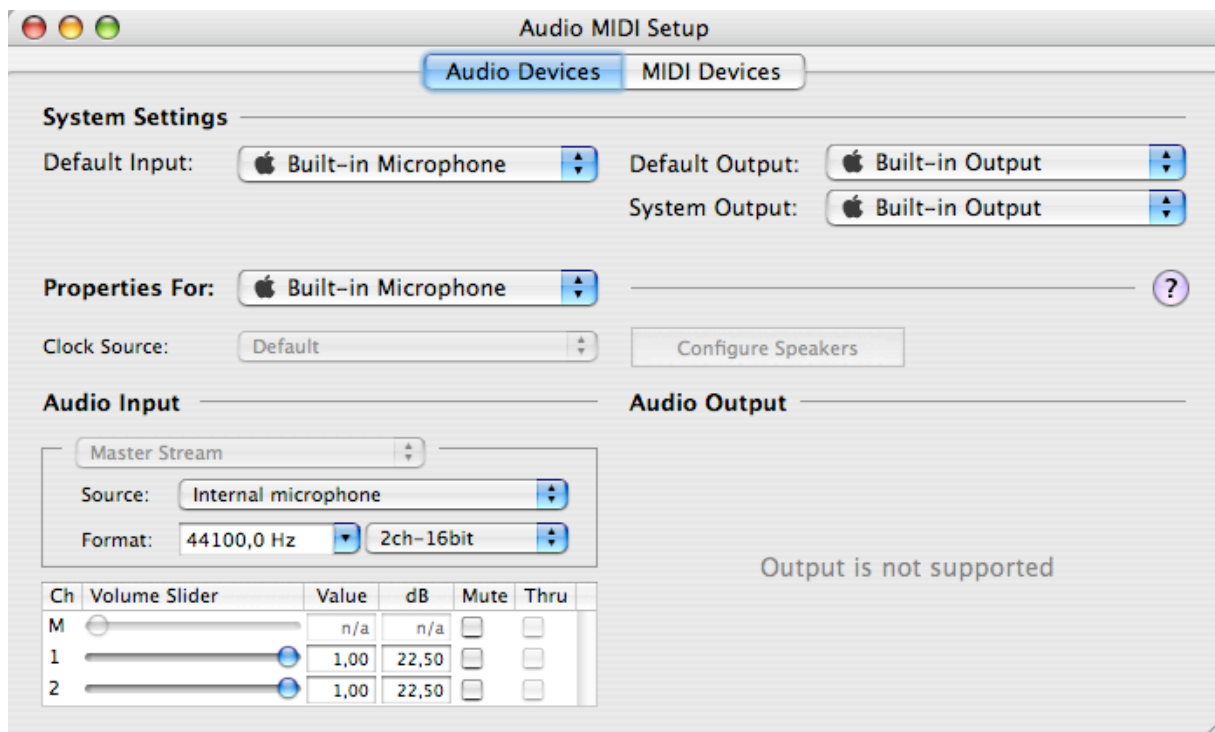




I/O Routing button

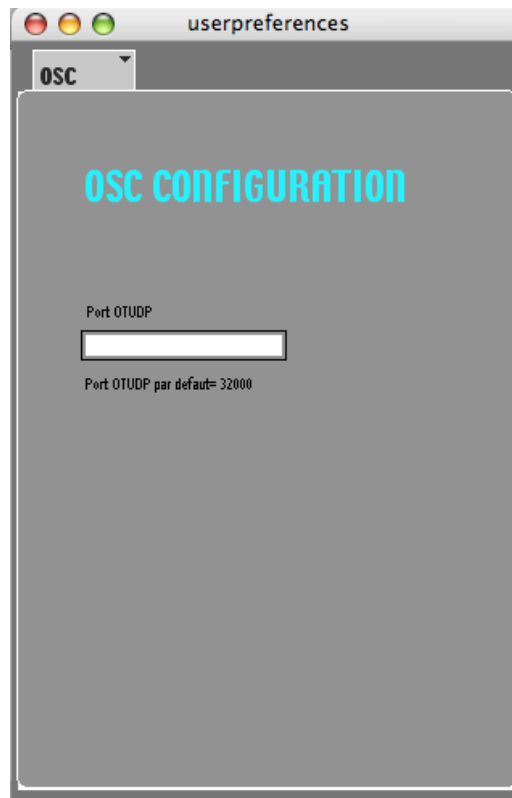


Driver Setup button (Mac os X Core Audio and Midi Setup)





## OSC Preferences (network)



### Description

You can send from an other computer data to the Peralvino Engine (i.e. from a video, lights specific computer with UDP capability). To respect the Peralvino Protocol

Default port is 32000.

## Menu bar functionalities

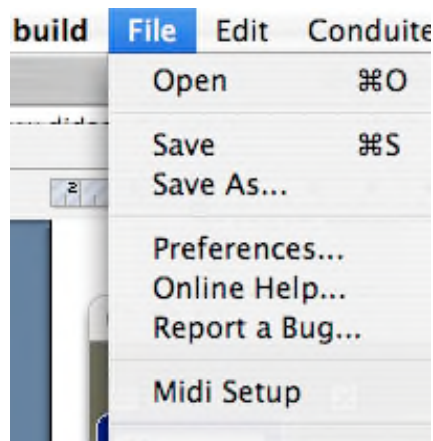


### Description

- Menu File
- Menu Edition
- Menu Conduite
- Menu Affichage
- Menu Sorties
- Menu Preferences
- Menu Window



## FILE Menu

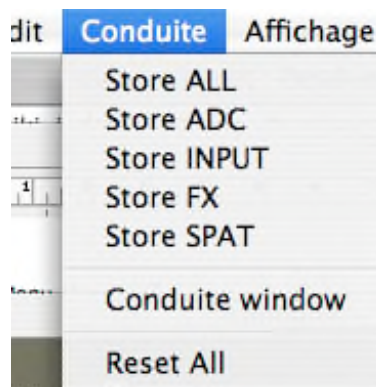


Open **not used**  
Save **not used**  
Save as **not used**

Preferences... : Open Preferences window (midi, keys, Dsp, OSC)  
Online Help... : Link to the Online documentation  
Report a Bug... : Link to User Forum

Midi setup : Open Max Midi port configuration window

## Conduite Menu



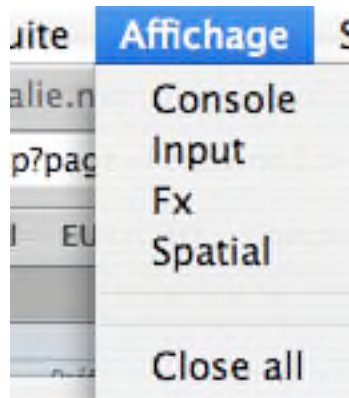
Store All : store ALL parameters  
Store Input : store Input page and modules parameters  
Store Fx : store FX page and parameters  
Store Spat : store Spatial page and parameters  
Store ADC : store ADC parameters

Conduite window : Open the Conduite window

Reset All : initialize All parameters, pages



## Affichage Menu



Console : Open Main console window

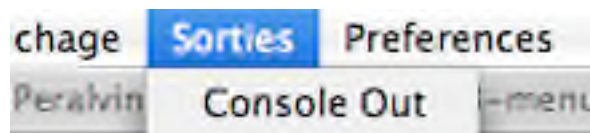
Input : Open Input page

Fx : Open Fx page

Spat : Open Spatial page

Close all : Close all pages

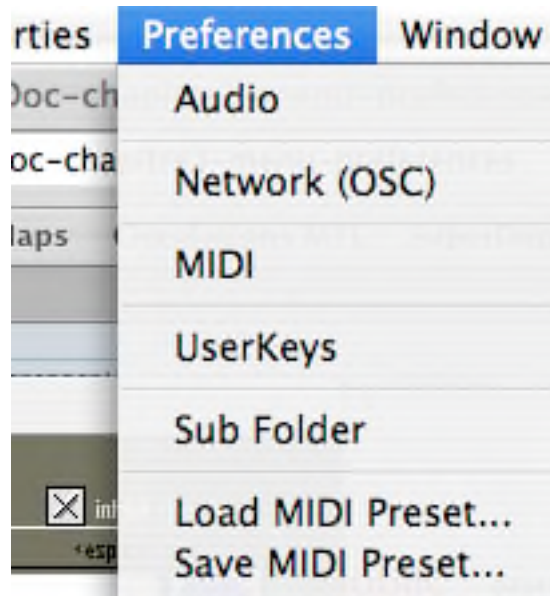
## Sorties Menu



Console OUT : Open Output console window



## Preferences Menu



Audio : Open preferences DSP window

Midi : Open preferences Midi window

[UserKeys](#) : Open User Key preferences window

OSC : Open OSC preferences window

SUB Folder : To manage all paths with Peralvino

Load MIDI preset : Load MIDI controller preset

Save MIDI preset : Save MIDI controller preset



## CHAPITRE 5

### INPUT Modules

#### SFmono



##### Description

SFmono module is a simple sound file player (Max/MSP sfplay~). This a monophonic module and you can use AIFF, SD2, WAVE sound file format.

##### Interface Parameters

gain : gain  
volume : output level of the Console input track  
loop on/off : enable/disable loop playback  
speed : speed playback (pitch shift)  
play/stop  
mute  
help : Open help window for this module

##### Script (#1=i1...i8)

gain #1-gain <0-infinity>  
volume #1-vol <0-1.27> 1=0 dB / Dynamic **Scripting** : #1-vol 1.08 3000 means #1-vol go to 1.08 in 3sec.  
loop #1-v3 <0,1>  
speed #1-v4 <0-infinity> 1=normal speed  
play/stop #1-play <0,1>  
mute #1-mute <0,1>  
#1-bloc cmd message: to send to sfplay~ object using a command message



## SF2 Module



### Description

SFmono module is a simple sound file player (Max/MSP sfplay~). This a stereo module and you can use AIFF, SD2, WAVE sound file format.

### Interface Parameters

gain : gain  
 volume : output level of the Console input track  
 loop on/off : enable/disable loop playback  
 speed : speed playback (pitch shift)  
 play/stop  
 mute  
 help : Open help window for this module

### Script (#1=i1...i8)

gain #1-gain <0-infinity>  
 volume #1-vol <0-1.27> 1=0 dB / Dynamic **Scripting** : #1-vol 1.08 3000 means #1-vol go to 1.08 in 3sec.  
 loop #1-v3 <0,1>  
 speed #1-v4 <0-infinity> 1=normal speed  
 play/stop #1-play <0,1>  
 mute #1-mute <0,1>  
 #1-bloc cmd message: to send to sfplay~ object using a command message

## SF4 module





### Description

SFmono module is a simple sound file player (Max/MSP sfplay~). This a 4 interlaced tracks module and you can use AIFF, SD2, WAVE sound file format.

### Interface Parameters

gain : gain  
volume : output level of the Console input track  
loop on/off : enable/disable loop playback  
speed : speed playback (pitch shift)  
play/stop  
mute  
help : Open help window for this module

### Script (#1=i1...i8)

gain #1-gain <0-infinity>  
volume #1-vol <0-1.27> 1=0 dB / Dynamic **Scripting** : #1-vol 1.08 3000 means #1-vol go to 1.08 in 3sec.  
loop #1-v3 <0,1>  
speed #1-v4 <0-infinity> 1=normal speed  
play/stop #1-play <0,1>  
mute #1-mute <0,1>  
#1-bloc cmd message: to send to sfplay~ object using a command message

## NoiseGen module



### Description

NoiseGen module is a white or pink Noise generator.

### Interface Parameters

gain : gain module  
volume : level  
Noise type : off, white, pink  
Mute  
help : help window

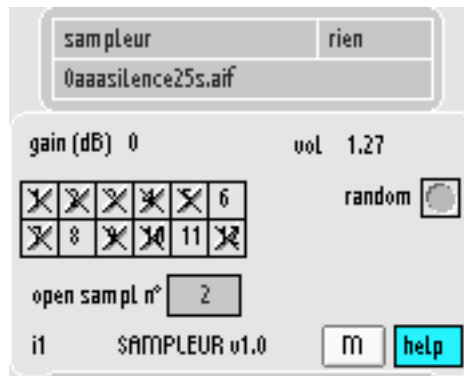
### Script (#1=i1...i8)

gain #1-gain <0-infinity>  
volume #1-vol <0-1.27> 1=0 dB  
Noise type #1-v1 <0, 1, 2> 0=off, 1=white, 2=pink  
mute #1-mute <0,1>





# Sampleur Module



## Description

Sampler module is a 12 **mono** sound file player. You can trigger all players using toggle interface or randomly.

## Interface Parameters

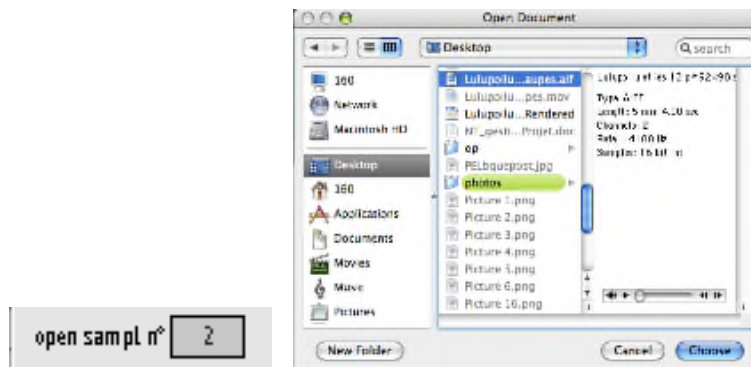
gain : amplitude

volume : level

mute

help : help window

Open sample n° : Open Sampl n° allow you to load a sound file into player n°



Play sample 1 to 12 : trigger playback

Random : bang a random playback

## Script (#1=i1...i8)

gain #1-gain <0-infinity>

volume #1-vol <0-1.27> 1=0 dB

loop #1-v3 <0,1>

mute #1-mute <0,1>

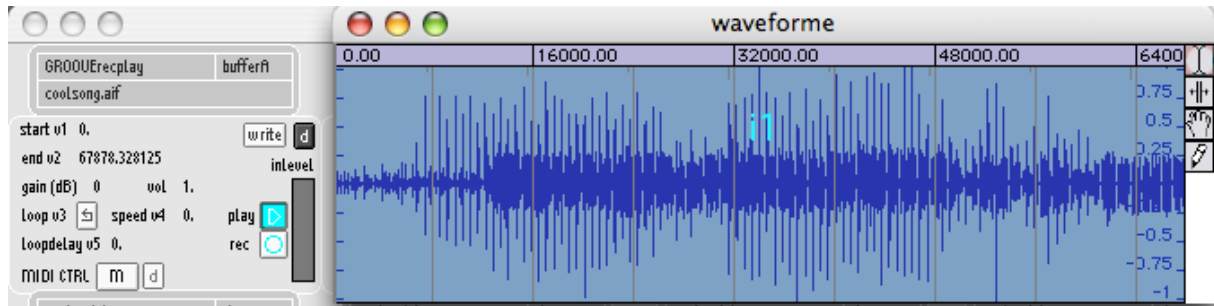
Open sample n° #1-selsampl op<1-12> <audio file name> / i.e. : i1-selsampl op2  
terrorista.wav

Play sample #1-pl<1-12> <0,1> / i.e. : i1-pl2 1

Random #1-ran <bang>



# GROOVErecplay module



## Description

GROOVErecplay module is a player/recorder (Max/MSP record~). This is a mono (AIFF) input module using buffer capabilities for recording an audio signal in Ram.

You have to assign a buffer (bufferA, B...H) after selecting this module. Buffer A is used to record a sound file and export it to your Hard Drive.

## Interface Parameters

gain : amplitude

volume : volume

start : Loop in or Start point for playback

end : Loop out or End point for playback

loop on/off : enable/disable loop playback

loopdelay : When loop is on, you can play loops with a delay between successive playback

speed : speed playback (pitch shift)

play/stop

rec

d button (display) : open the buffer waveform window

meter : GROOVErecplay input meter

write button : export a buffer to Disk

## Script (#1=i1...i8)

gain #1-gain <0-infinity>

volume #1-vol <0-1.27> 1=0 dB

start #1-v1 <0.-maximum size of buffer> in samples

end #1-v2 <0.-maximum size of buffer> in samples

loop #1-v3 <0,1>

speed #1-v4 <0.-infinity> 1=normal speed playback

loopdelay #1-v5 <0., infinite> 0 = loop without delay

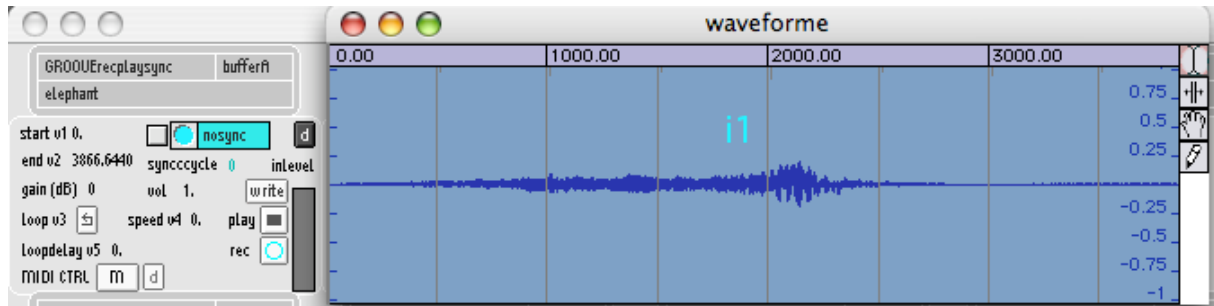
play/stop #1-play <0,1>

rec #1-rec <0,1>

#1-view <0,1> Open the waveform window



# GROOVErecplaysync module

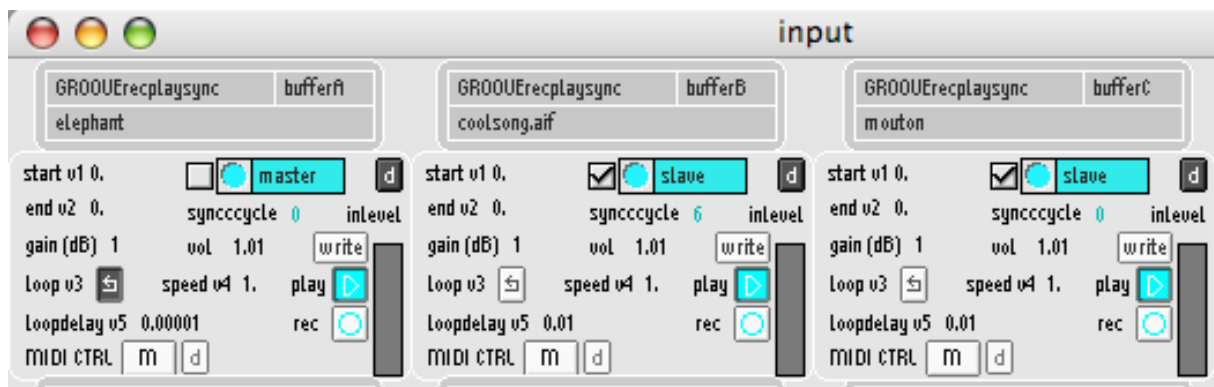


## Description

GROOVErecplaysync is a mono player/recorder module.

SYNC function allow you to synchronize more than two GrooveRecplaySYNC module together. One of them is the Master, the others GROOVErecplaysync are Slaves.

Example, How to :



i1 GROOVErecplaysync (linked to Buffer A) is the master Sync reference, i2 (Buffer B) and i3 (Buffer C) are Slaves. Enable syncmode on i2 and i3 with the toggle. Setup the syncycle value. This parameter is the length of duration between start point and end point of playback for Slave modules based on the Master Sync duration reference (1 cycle is the length of the sound file Master module). Setup speed, gain, volume parameters. And Enable play button. At the end of the sound file Master module, all others Slaves modules are triggered depending on the syncycle value.

## User Interface

gain : amplitude  
 volume : level  
 start : loop in point  
 end : loop out point  
 loop on/off : Enable/disable loop mode  
 speed : playback speed (pitch shift)  
 loopdelay : delay between two loops  
 rec  
 d button (display) : Open buffer waveform window  
 meter : input module meter  
 sync : select sync mode (off, master, slave)  
 syncycleenable : enable the slave sync triggering



synccycle : number of cycle for playback in sync Slave mode

write button : export a buffer to Disk

## Script (#1=i1...i8)

gain #1-gain <0-infinity>

volume #1-vol <0.-1.27> 1=0 dB

start #1-v1 <0.-maximum size of buffer> in samples

end #1-v2 <0.-maximum size of buffer> in samples

loop #1-v3 <0,1>

speed #1-v4 <0.-infinity> 1=normal speed

loopdelay #1-v5 <0., infinite> 0 = loop without silence

play/stop #1-play <0,1>

rec #1-rec <0,1>

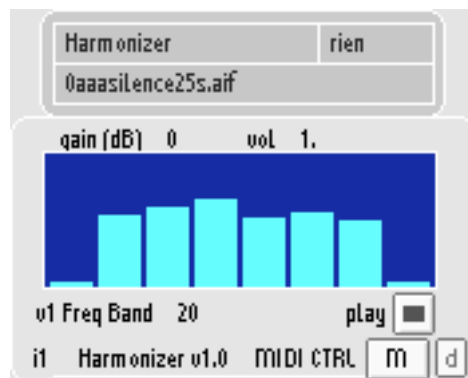
sync #1-sync <0,1,2> 0=nosync, 1=master, 2=slave

synccycleenable #1-synccycleenable <0,1>

synccycle #1-synccycle <0-infinity>

#1-view <0,1> Open/Close the buffer waveform window

## Harmonizer module



### Description

Harmonizer module is a 8 harmonics frequencies generator.

### Interface Parameters

gain : amplitude

volume : level

Freq band : fundamental frequency and harmonics offset (i.e.: if Freq Band=60, h2=120, h3=180...)

play

Frequency Gain window : level of Fundamental and Harmonic frequencies

### Scripting (#1=i1 to i8)

gain #1-gain <0,16>

volume #1-vol <1-1.27>

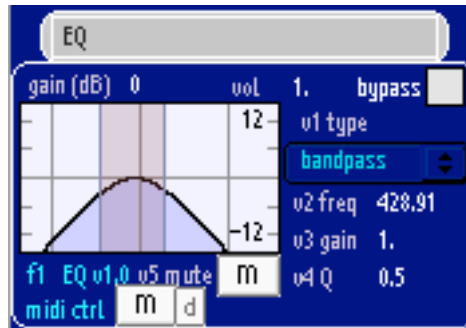
Freq Band #1-v1 <20-1000> in Hz

Frequencies Gain #1-v2 <list of 8 float>, gain <0.,1.>



# BLOC FX

## EQ module



### Description

EQ is a simple parametric equalizer (biquad~).

### Interface Parameters

gain : amplitude

volume : level

type : type

freq: main frequency

gain : frequency amplitude level

Q : bandwidth, filter acuity

Mute

help : open the help window

### Script (#1=f1...f8)

gain #1-gain <0-infinity>

volume #1-vol <0-1.27> 1=0 dB

type #1-v1 < 0-7> 1 = low pass, 2 = high pass, 3 = band pass, 4 = band stop, 5 = peak notch, 6 = low shelf, 7 = high shelf

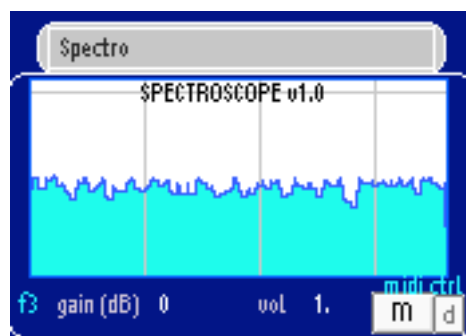
freq #1-v2 <20., 20000.>

gain #1-v3 <0.-20.> 1. = no gain, 1 to 20 = trim up to, 0. to 1. = trim down to

Q #1-v4 <0.5, 25.>

mute #1-mute <0,1>

## Spectro module





## Description

Spectro is a basic spectroscope. Visualization from 20 to 20000 Hz.

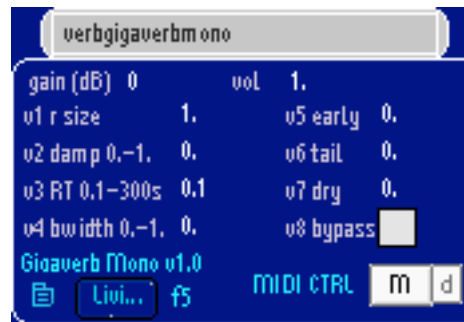
## Interface Parameters

gain : amplitude  
 volume : level input  
 help : Open the help window

## Script (#1=f1...f8)

gain #1-gain <0-infinity>  
 volume #1-vol <0-1.27> 1=0 dB  
 mute #1-mute <0,1>

# Gigaverbmono module



## Description

Gigaverb is a mono reverb processor based on gigaverb~ object

## Interface Parameters

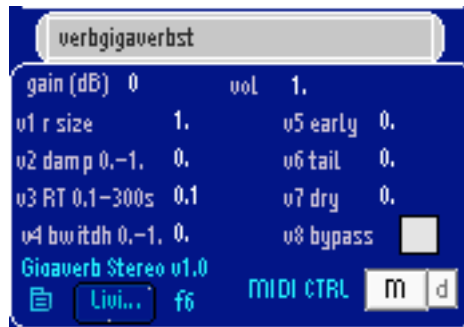
gain : amplitude  
 volume : level  
 size : size of the hall  
 damp : High/Low frequency ratio  
 RT : Reverberation Time  
 bandwidth : bandwidth  
 early : Early reflections level  
 tail : Tail Reverb level  
 dry : Direct field level  
 bypass  
 help : Open help window

## Script (#1=f1...f8)

gain #1-gain <0-infinity>  
 volume #1-vol <0-1.27> 1=0 dB  
 size #1-v# <1. - 300.> in meters  
 damp #1-v2 <0.-1.>  
 RT #1-v3 <0.1-300> in seconds  
 bandwidth : #1-v4 <0.-1>  
 early : #1-v5 <0.-1>  
 tail : #1-v6 <0.-1>  
 dry : #1-v7 <0.-1>  
 bypass : #1-v8 0,1



# Verbstereo module



## Description

VerbStereo is a stereo version of GigaVerbmono.

## Interface Parameters

Same as GigaVerbmono

## Script (#1=f1...f8)

Same as GigaVerbmono

# MonoVerb module



## Description

Monoverb is a reverb processor based on monoverb~ object.

## Interface Parameters

gain : amplitude

volume : level

size : size of the hall

damp : high/low frequency ratio

wet : Reverb field level

dry : Direct field level

freeze : reverb sound freeze function

bypass

help : open the help window

## Script (#1=f1...f8)

gain #1-gain <0-infinity>

volume #1-vol <0-1.27> 1=0 dB

size #1-v1 <0.-1.> in meter



damp #1-v2 <0.-1.>  
wet : #1-v3 <0.-1.>  
dry : #1-v4 <0.-1.>  
freeze : #1-v5 <0-1>  
bypass : #1-v8 <0-1>

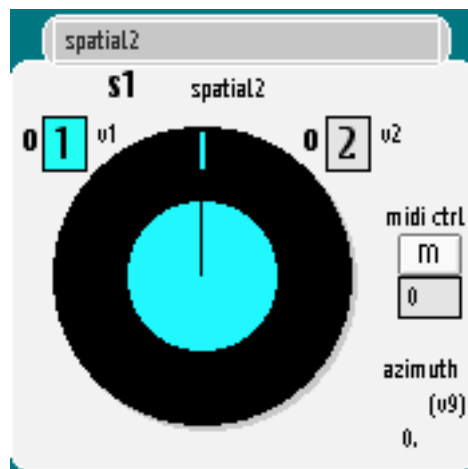
## Credits

monoverb~ version 1.0 - Max 4.2 (OS 9 & OS X)

by Olaf Matthes - By using this software you are agreeing to the software license include in the application folder. This software is copyright © 2003 by Olaf Matthes. All rights reserved.

# BLOC SPAT

## spatial2 module



## Description

Spatial2 is a stereo pan pot based on rvbap object.

## Parameters

o1 and o2 :Direct output routing (to the dac)  
Pan pot : azimuth

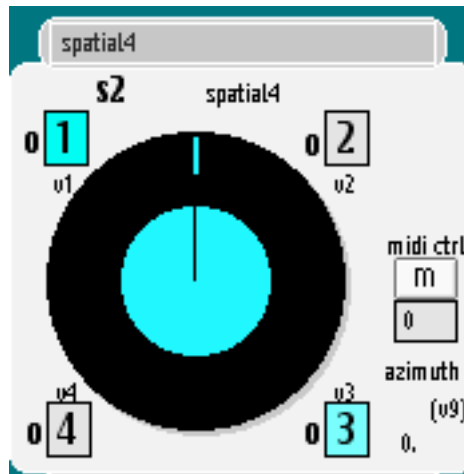
## Script (#1=s1...s8)

routing o1 : #1-v1 <1-8>  
routing o2 : #1-v2 <1-8>  
azimuth : #1-v9 <-36000-36000>





## Spatial4 module



### Description

Spatial4 is a 4 outputs pan pot based on rvbap object.

### Parameters

o1 to o4 :Direct output routing (to the dac)

Pan pot : azimuth

### Script (#1=s1...s8)

routing o1 : #1-v1 <1-8>

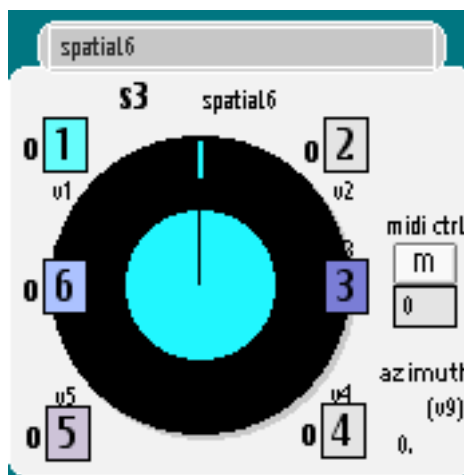
routing o2 : #1-v2 <1-8>

routing o3 : #1-v3 <1-8>

routing o4 : #1-v4 <1-8>

azimuth : #1-v9 <-36000-36000>

## Spatial6 module



### Description

Spatial6 is a 6 outputs pan pot based on rvbap object.



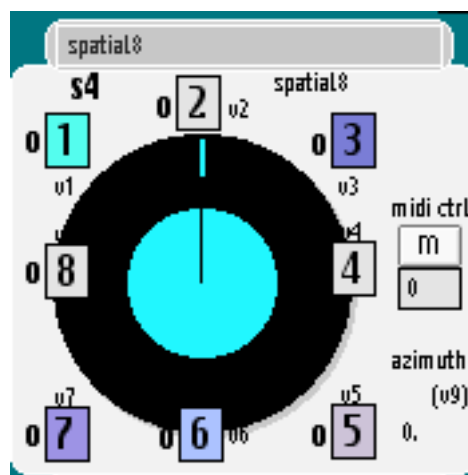
## Parameters

o1 to o6 :Direct output routing (to the dac)  
Pan pot : azimuth

## Script (#1=s1...s8)

```
routing o1 : #1-v1 <1-8>  
routing o2 : #1-v2 <1-8>  
routing o3 : #1-v3 <1-8>  
routing o4 : #1-v4 <1-8>  
routing o5 : #1-v5 <1-8>  
routing o6 : #1-v6 <1-8>  
azimuth : #1-v9 <-36000-36000>
```

# Spatial8 module



## Description

Spatial8 is a 8 outputs pan pot based on rvbap object.

## Parameters

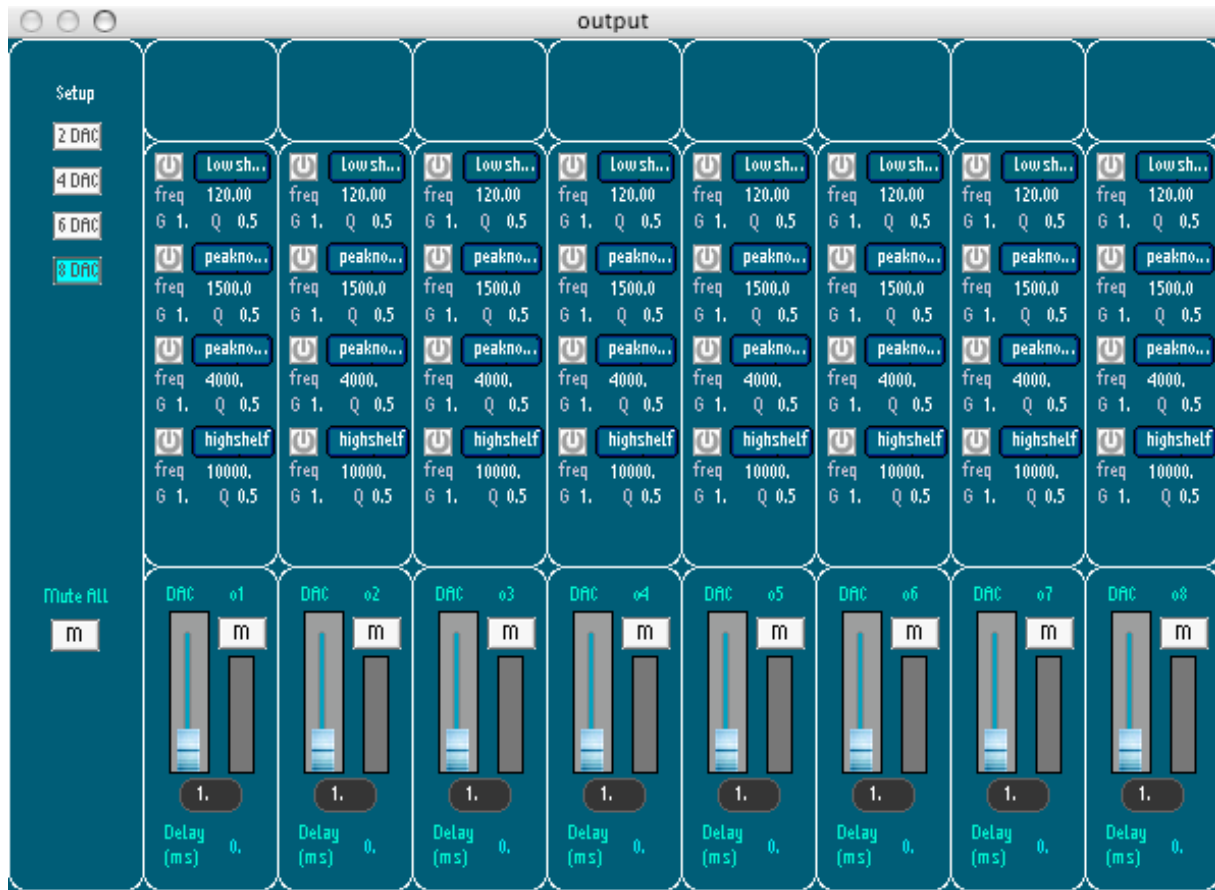
o1 to o8 :Direct output routing (to the dac)  
Pan pot : azimuth

## Script (#1=s1...s8)

```
routing o1 : #1-v1 <1-8>  
routing o2 : #1-v2 <1-8>  
routing o3 : #1-v3 <1-8>  
routing o4 : #1-v4 <1-8>  
routing o5 : #1-v5 <1-8>  
routing o6 : #1-v6 <1-8>  
routing o7 : #1-v7 <1-8>  
routing o8 : #1-v8 <1-8>  
azimuth : #1-v9 <-36000-36000>
```



# Output CONSOLE



## Script

(#1 = o1 to o8)

output-setup <2,4,6,8> number of outputs

#1-level <0,1> level of track #1

#1-mute <0,1> mute of track #1

#1-delay <0.-infini> Delay in milliseconds

EQ (eq 1 is the first upper one)

#1-eq1switch <0,1> Enable/disable Eq1

#1-eq2switch <0,1> Enable/disable Eq2

#1-eq3switch <0,1> Enable/disable Eq3

#1-eq4switch <0,1> Enable/disable Eq4

Eq 1:

#1-v1 <0,7> type of filter: 0=display, 1=low pass, 2=high pass, 3=band pass, 4=band stop, 5=peak notch, 6=low self, 7=high self

#1-v2 <20.,20000.> frequency

#1-v3 <0.,24.> gain

#1-v4 <0.5,16.> filter quality in octave

Eq 2:

#1-v5 <0,7> type of filter: 0=display, 1=low pass, 2=high pass, 3=band pass, 4=band stop, 5=peak notch, 6=low self, 7=high self



#1-v6 <20.,20000.> frequency  
 #1-v7 <0.,24.> gain  
 #1-v8 <0.5,16.> filter quality in octave

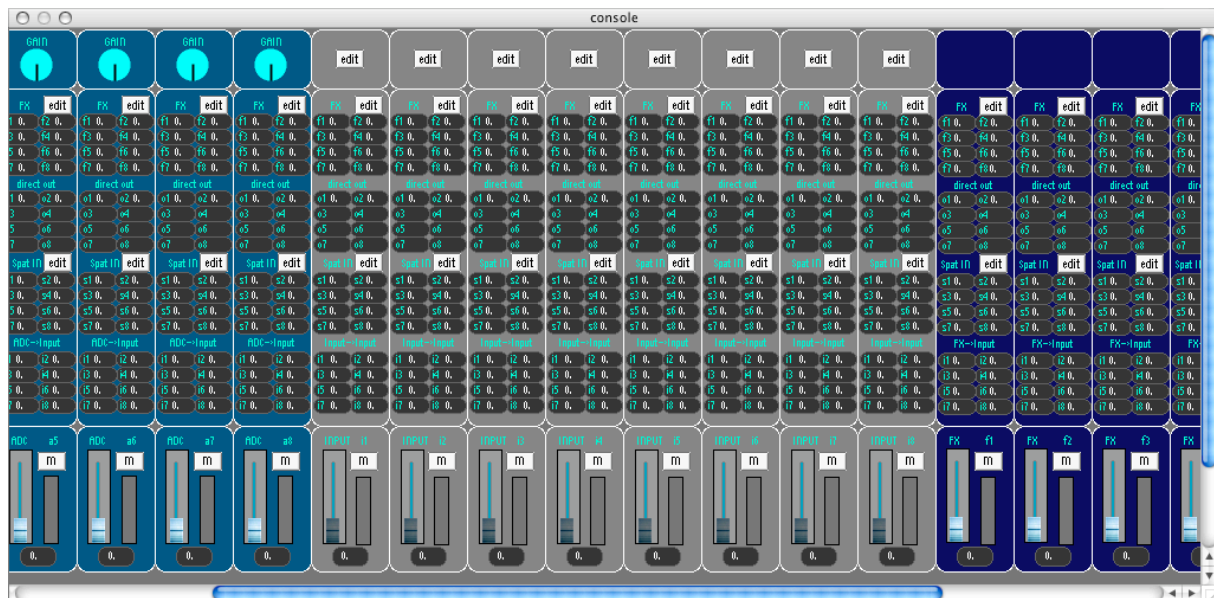
Eq 3:

#1-v9 <0,7> type of filter: 0=display, 1=low pass, 2=high pass, 3=band pass, 4=band stop, 5=peak notch, 6=low self, 7=high self  
 #1-v10 <20.,20000.> frequency  
 #1-v11 <0.,24.> gain  
 #1-v12 <0.5,16.> Filter quality in octave

Eq 4:

#1-v13 <0,7> type of filter: 0=display, 1=low pass, 2=high pass, 3=band pass, 4=band stop, 5=peak notch, 6=low self, 7=high self  
 #1-v14 <20.,20000.> frequency  
 #1-v15 <0.,24.> gain  
 #1-v16 <0.5,16.> filter quality in octave

## MAIN CONSOLE



### Script

ADC tracks Parameters :  
 (#1 = a1 to a8)

#1-gain <0,16> input trim  
 #1-vol <0.,1.27> track level

#1->o1...o8 <0.,1.> #1 to direct output 1 to 8  
 #1->f1...f8 <0.,1.> #1 to FX module 1 to 8  
 #1->s1...s8 <0.,1.> #1 to spatial module 1 to 8  
 #1->i1...i8 <0.,1.> #1 to input module 1 to 8



INPUT module Parameters :  
(#1 = i1 to i8)

#1-vol <0.,1.27> level track

#1->o1...o8 <0.,1.> #1 to direct output 1 to 8  
#1->f1...f8 <0.,1.> #1 to FX module 1 to 8  
#1->s1...s8 <0.,1.> #1 to spatial module 1 to 8  
#1->i1...i8 <0.,1.> #1 to input module 1 to 8

FX module Parameters :  
(#1 = f1 to f8)

#1-vol <0.,1.27> level track

#1->o1...o8 <0.,1.> #1 to direct output 1 to 8  
#1->f1...f8 <0.,1.> #1 to FX module 1 to 8  
#1->s1...s8 <0.,1.> #1 to spatial module 1 to 8  
#1->i1...i8 <0.,1.> #1 to input module 1 to 8

## MIDI MAPPING - MIDI Controller

### Description

You can assign Midi Controller to Peralvino parameters.

### How to ?

Plug your MIDI Devices

Setup the MIDI preference with your Device

If your device doesn't appears, click on 'Refresh'.



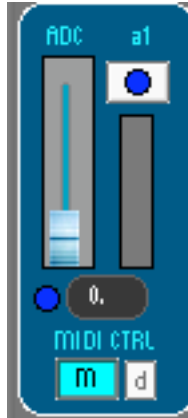
MIDI LAYER : allow you to assign MIDI controller. The shortcut to this layer is shift+m (M).

Open the Main console window, and press shift+m



## User Manual

---

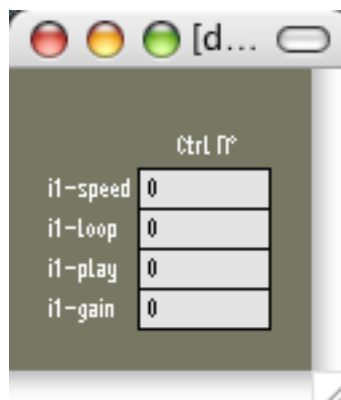


Blue round shows available parameters to map with MIDI controller.

You select by clicking on a blue round a parameter, the blue round become cyan. Change a midi slider on your Device, that's all. This an auto-learn function.

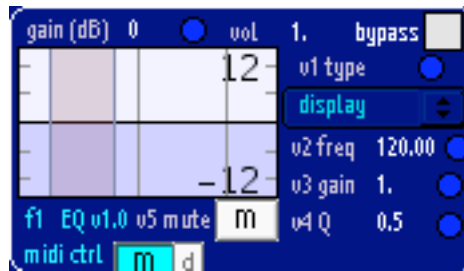


Viewing midi mapping information, click on the button 'd' below MIDI-CTRL





## Available MIDI parameters in FX module :



## Available MIDI parameters in Input module :

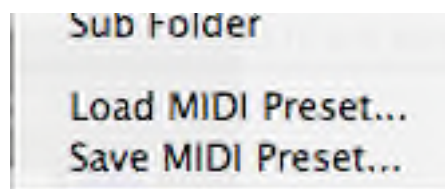


## Available MIDI parameters in Output Console :



## Save and Load MIDI mapping configuration

Menu->Preferences



To save a preset: 'Save MIDI Preset'

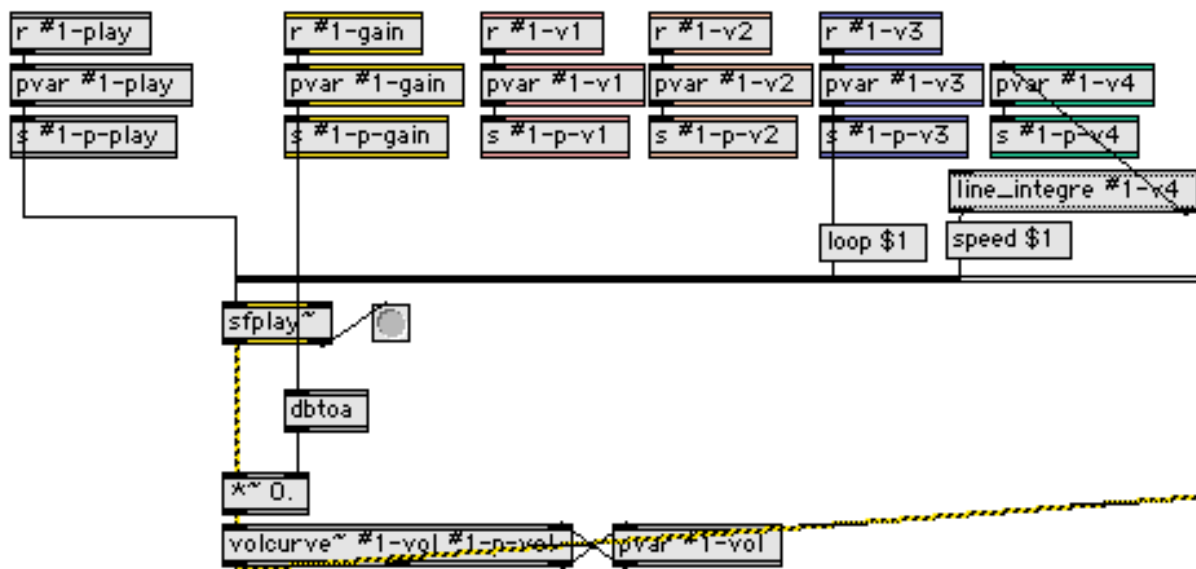
To load a preset: 'Load MIDI Preset'



## CHAPTER 6

# Addresses - Scripting - Protocol

Peralvino is an engine which can receive command from scripted message.



Protocol

Static addresses :

address name - variable; (i.e. : i1-gain 0.47 setup the gain to 0.47)

dynamic addresses :

address name - variable1 - variable2; (i.e. : i1-vol 1.15 3000 setup the level from a previous state to 1.15 in 3 seconds)

Some command line have to follow sequential rules :

i1-selblock SFmono; first choose a Sfmmono in slot1

i1-selbuffer rien; second choose 'rien' in buffer menu

i1-selsong sanantonio.aif; third choose a sound file

i1-gain 0; etc...

i1-v3 1;

i1-v4 1;

i1-vol 0;

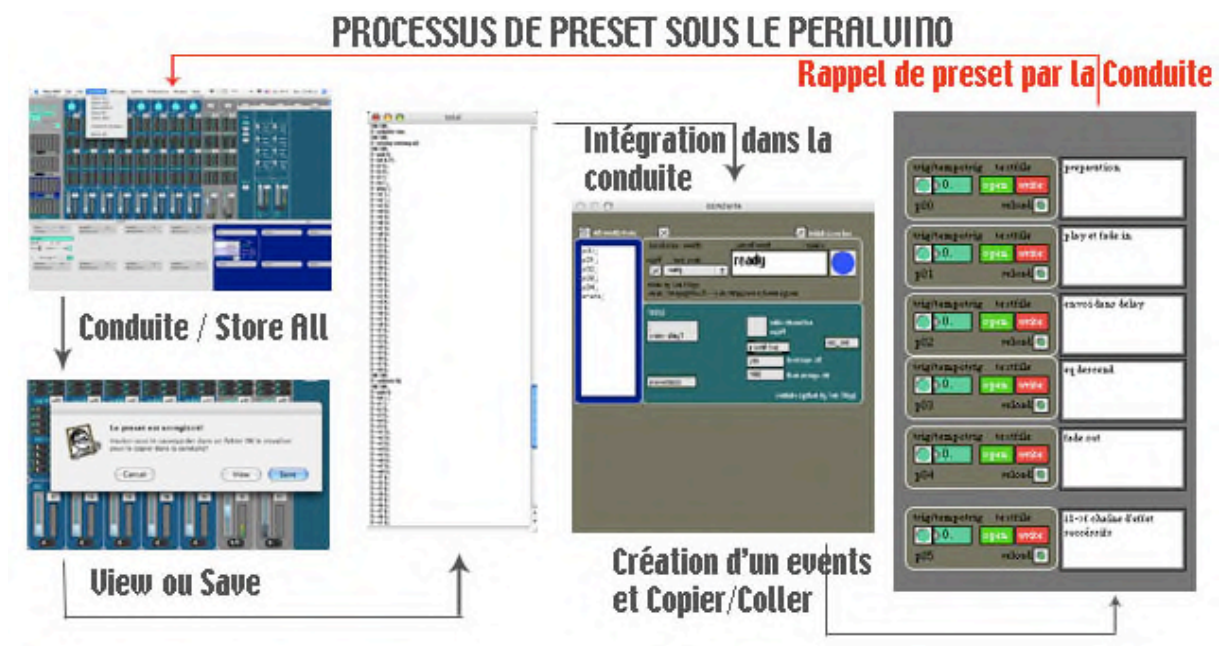
i1-play 0;

If these rules are not respected, Peralvino won't be setup correctly.





# Preset Architecture



## Description

### Storing presets

We have a long discussion on this topic. The final decision was to design an architecture based on script. Request a preset with menu Conduite->Store All generate a text which collect all static parameters (like a picture of the interface). This scripted picture is contained into Events of cue list module by copy/paste operation.

### Recalling presets

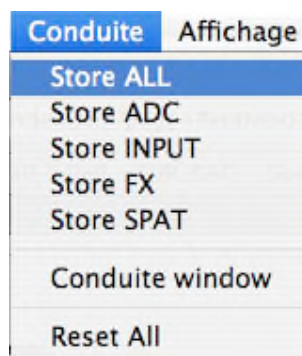
Recalling a preset is done in the Conduite window.

### Editing presets:

You can modify all scripted line in the events window and replace a text file to save a cue.

## How to ?

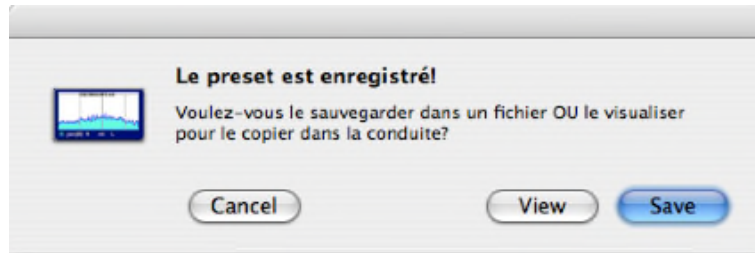
To save (generate) a preset:  
Menu bar Conduite->Store All



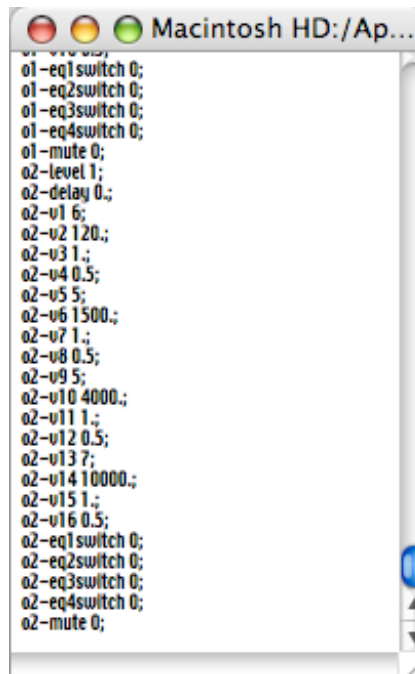


## User Manual

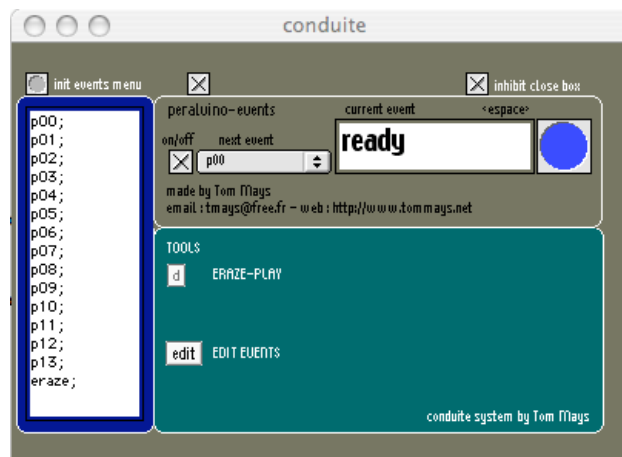
A pop-up window appears. You'll be asked to save or view the preset as a text, click on view.



Save : save a text file on HD to use it later !



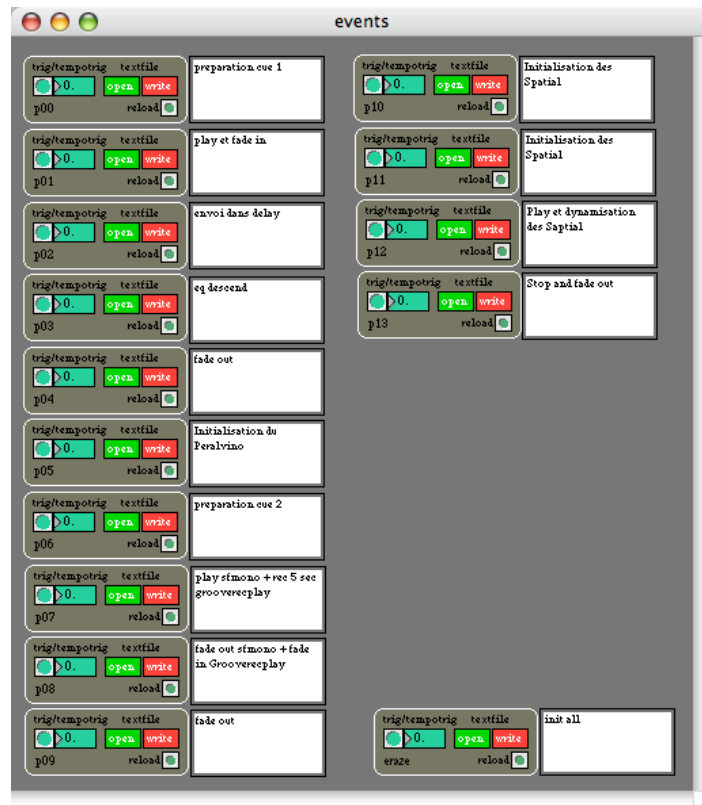
At this step, Conduite window must be open. We have to copy/paste the scripted preset.



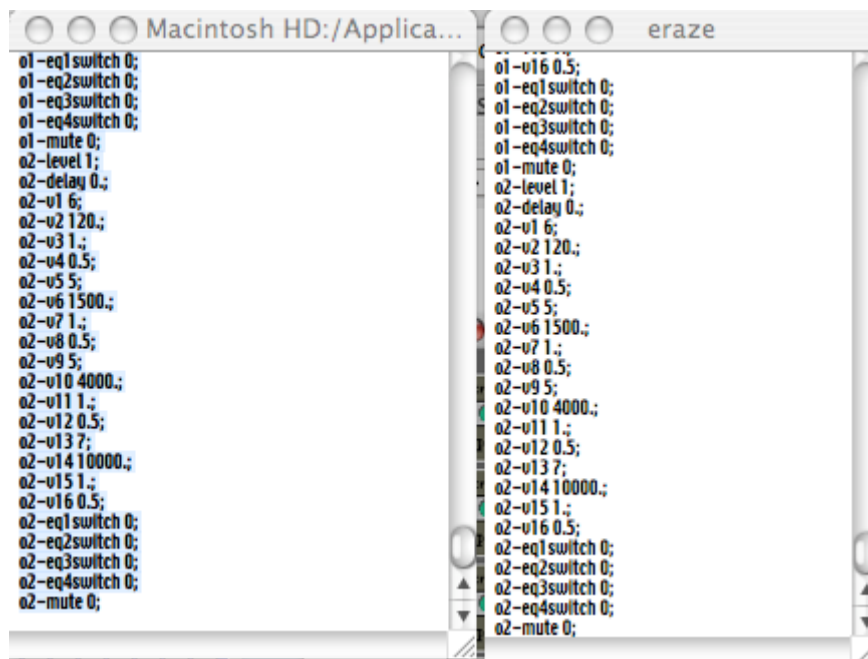
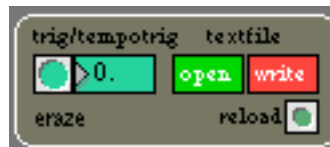
Click EDIT EVENTS button :



# User Manual



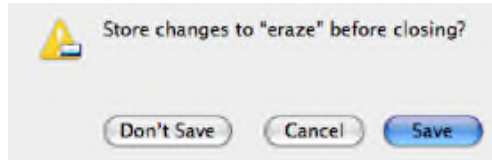
Open an Event, Click on "open" button, select all text and delete it, and copy the new preset.





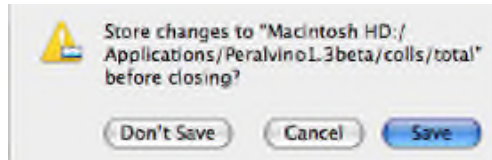
## User Manual

---



Close the event text window and Save changes before closing. Your new preset is now in the event, ready to be triggered by the Conduite window.

Close text window.





## CHAPTER 7

### New modules - Programmation

**Under construction**



# References and links

web :

[http://www.didascalie.net/tiki-index.php?page=Peralvino\\_zone](http://www.didascalie.net/tiki-index.php?page=Peralvino_zone)

Users Forum :

[http://www.didascalie.net/tiki-view\\_forum.php?forumId=21](http://www.didascalie.net/tiki-view_forum.php?forumId=21)

Online Documentation :

[http://www.didascalie.net/tiki-index.php?page=peralvino\\_documentation](http://www.didascalie.net/tiki-index.php?page=peralvino_documentation)

Credits :

Peralvino is Under GNU/GPL © 2006 – Olivier PFEIFFER / Guy LEVESQUE