

Yan Maresz

Tutti

2013

Concert Lausanne 13th March 2023
2024



The setup and the execution of the electroacoustic part
of this work requires a Computer Music Designer (Max expert).

Table of Contents

Table of Contents	2
Work related information	3
Performance details	3
Detailed staff	3
Realisation	3
Useful links on Brahms	3
Version related information	4
Documentalist	4
Realisation	4
Upgrade Motivation	4
Other version(s)	4
Electronic equipment list	5
Computer Music Equipment	5
Files	6
Instructions	7
Software Installation	7
General Setup	7
Loudspeaker setup	7
Stage setup	8
Audio Setup	9
DSP status for both patches	9
Master Patch	9
PTRM Patch	9
Initialization routine	10
Shortcuts	10
Patch Presentation	10
Performance Notes	13
Simulation	13
Program note	14

Work related information

Performance details

- June 6, 2013, Paris, Centre Pompidou, Grande salle, festival ManiFeste

Publisher : Durand

Detailed staff

- flute (also piccolo), oboe, clarinet (also contrabass clarinet), bassoon, horn, trumpet [en ut] , tenor/bass trombone, tuba, percussionist, piano (also electronic/MIDI keyboard/synthesizer [MIDI]), violin, second violin, viola, cello, double bass

Realisation

- Thomas Goepfer

Useful links on Brahms

- [Tutti for ensemble \(2013\), 25mn](#)
- [Yan Maresz](#)

Version related information

Performance date: March 13, 2023

Documentation date: April 15, 2024

Version state: valid, validation date : May 20, 2024, update : May 20, 2024, 1:49 p.m.

Documentalist

yann brecy (yann.brecy@ircam.fr)

You noticed a mistake in this documentation or you were really happy to use it? Send us feedback!

Realisation

- Maxime Le Saux (Sound engineer)
- Thomas Goepfer (Computer Music Designer)

Version length: 25 mn

Default work length: 25 mn

Upgrade Motivation

Upgraded documentation for Max8 version

Other version(s)

- [Yan Maresz - Tutti - Concert Lausanne 13th March 2023-aeffacer \(March 14, 2023\)](#)
- [Yan Maresz - Tutti - Creation Cologne \(June 9, 2013\)](#)

Electronic equipment list

Computer Music Equipment

- 1 MacBook Pro - *Apple Laptops* (Apple)
- 1 iPad - *Tablets* (Apple)
with Lemur software or for control with Mira
- 1 Max 8 - *Max* (Cycling74)
in Rosetta Mode
- 1 Fireface 800 - *Sound Board* (RME)
8 inputs, 8 outputs

Files

File	Type	Author(s)	Comment
Technical rider Maresz.pdf	Technical rider		
Concert Patch	Patch	Benoit Meudic	password required (tutti2023)
Repet-130605 Project.dmg	Ableton session		

Instructions

This piece uses 2 different patches (on the same computer) : one for polyrythm machine (called PRTM), and another patch for electronic treatments. This second patch is the master patch, and control the other via local OSC.

Audio is sent and received via soundcard loopback or via your loopback solution of choice from one patch to another.

Audio from microphones sent to computers through mixing desk. A midi keyboard on stage is used to triggers events. Master and Spare computers receive de same inputs

Software Installation

Download and install Max 8.X.X

Download, open and copy the content of the Concert Patch archive (Tutti_Patchs-2023.dmg)

2 different instances of Max 8.X.X launched in rosetta mode is necessary

Put the whole folder in your max file preferences. Be sure to remove any other antescofo object from your path.

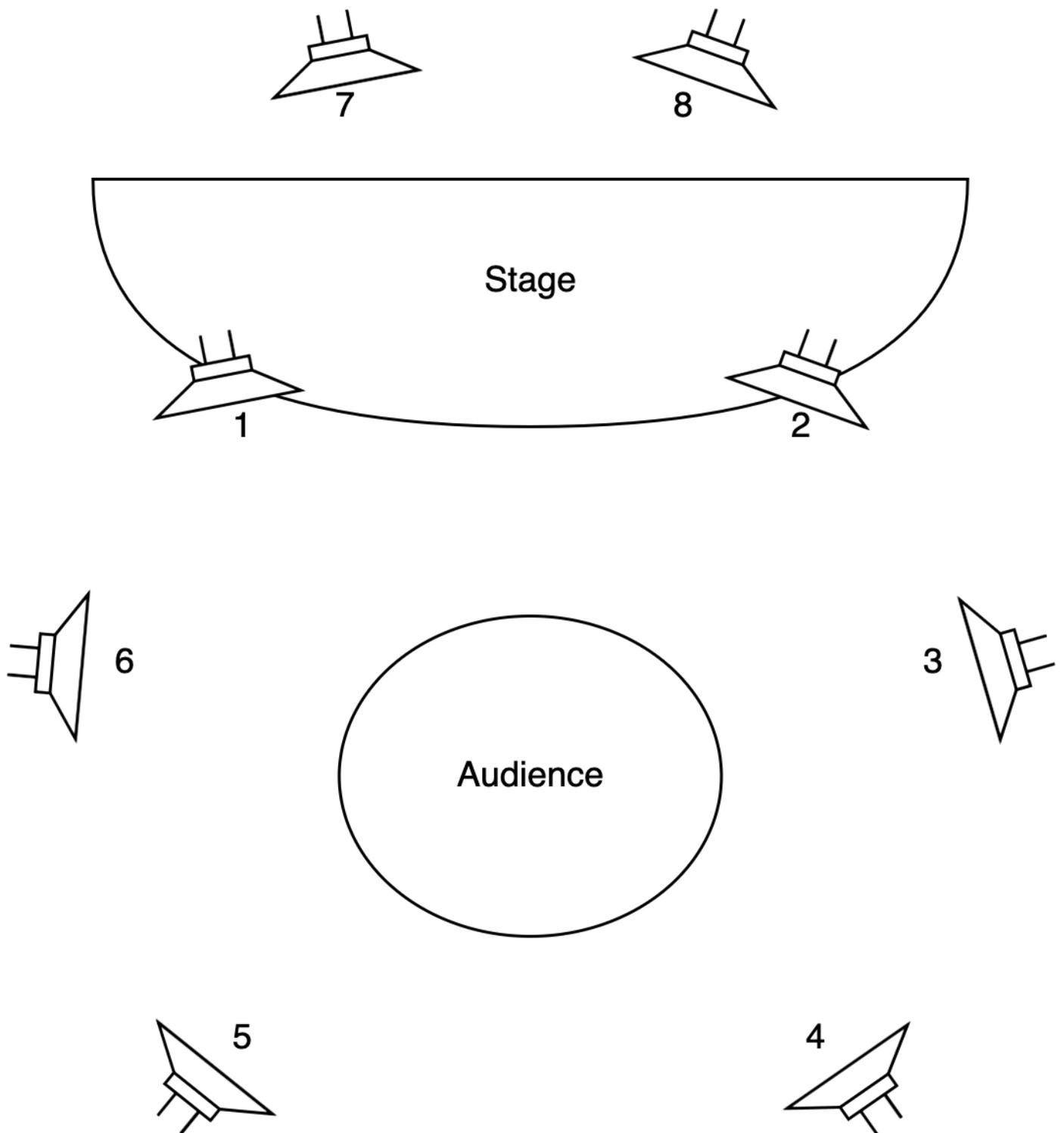
All the necessary packages and external are provided, but are intel based

- The PTRM is *AntePolyRyt.maxproj*
- The Master and mixing Patch is *Tutti-Master-1e.maxpat*

General Setup

Loudspeaker setup

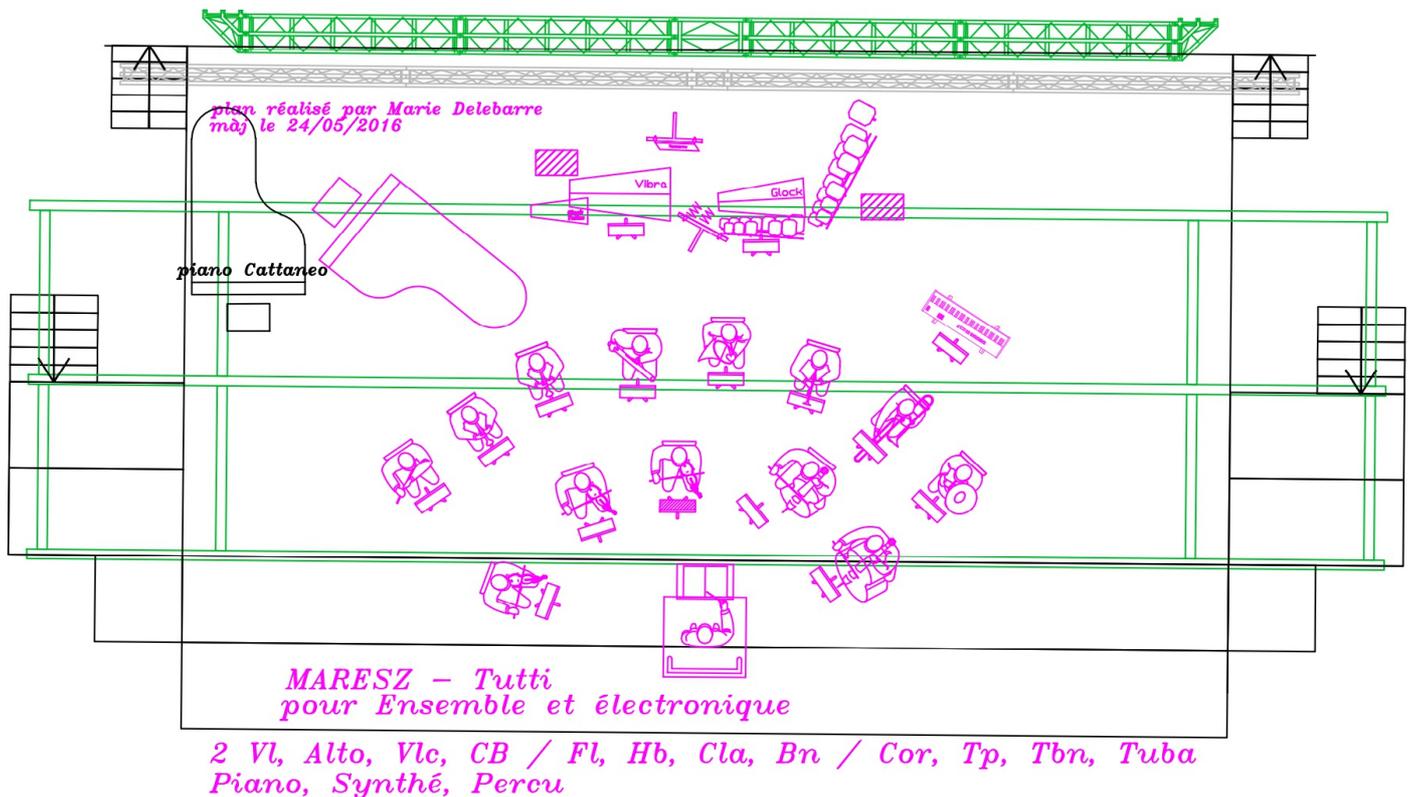
Back speakers are elevated to avoid to embarrass orchestra, they can be directed to wall.



- 1 : stage front, left
- 2 : stage front, right
- 3 : audience right
- 4 : audience back, right
- 5 : audience back, left
- 6 : audience left
- 7 : stage back, left
- 8 : stage back, right

The details of the loudspeaker setup are in the technical rider.

Stage setup



Audio Setup

Audio is sent between patches through loopback. Use hardware loopback with your soundcard or a software solution like an aggregate device with blackhole (<https://existential.audio/blackhole>)

DSP status for both patches

- SR : 44,1 kHz
- IO 512
- VS 512
- Overdrive ON
- Audio Interrupt OFF

Master Patch

- **4 audio input through loopback** : adc 5 to 8 from *AntePolyRyt.maxproj*
- **8 audio input** : adc 15 to 22
 - 15 : Orchestra Left
 - 16 : Orchestra Right
 - 17 : Piano
 - 18 : Violin 1
 - 19 : Violin 2
 - 20 : Viola
 - 21 : Cello
 - 22 : Double bass
- **8 audio output** : dac 15 to 22. To speakers
- **2 audio output** : dac 23 to 24. To orchestra and piano
- **2 audio output through loopback** : dac 3 to 4. To *AntePolyRyt.maxproj*

PTRM Patch

- **2 audio input through loopback** : adc 3 to 4. from *Tutti-Master-1e.maxpat*

- **4 audio output through loopback** : dac 5 to 8. to *Tutti-Master-1e.maxpat*

Initialization routine

- Open patch in one Max8 instance /AntPolyRyt-Score/AntePolytRyt/AntePolytRyt.maxproj
- Click “ALL.INIT bang” button
- start DSP

-
- Open patch in a second Max8 instance /Tutti-Main/PATCH_TUTTI_2016_MAX7/Tutti-MASTER-1e.maxpat
 - Check in “p BUFFERS” that media are loaded
 - if not load them (located in Tutti-Main/SOUND)
 - click RESET button
 - start DSP
 - click RESET button again
 - Load score to follow : *_Tutti.asco*
 - configure midi keyboard reception and activate midi reception (blinking button in (6))

At this point, the patch is ready and is waiting the first midi keyboard note to start. the main patch control the PTRM via local OSC

Mix and control is achieved via Mira web or LEMUR on Ipad. open and edit patcher “p MIDI/OSC”. Interface for Lemur software is in /Tutti-Main/PATCH_TUTTI_2016_MAX7/LEMUR/lemur1e.jzml

Shortcuts

If Keyboard shortcuts are on :

- Right arrow for next event
- Left arrow for previous event

Patch Presentation

Main Patch :

Grey “slider boxes” are monitors for each effect. Click on the effect name to pen effect interface. Name becomes green when effect is active. Names becomes green when effect is active. Vertical fader is a post fader, controlled by Lemur interface. M is the dB value of this post fader. O is the effect output dB, -127 means the treatment is off.

The screenshot shows the Mira web interface for audio processing. The interface is divided into several sections:

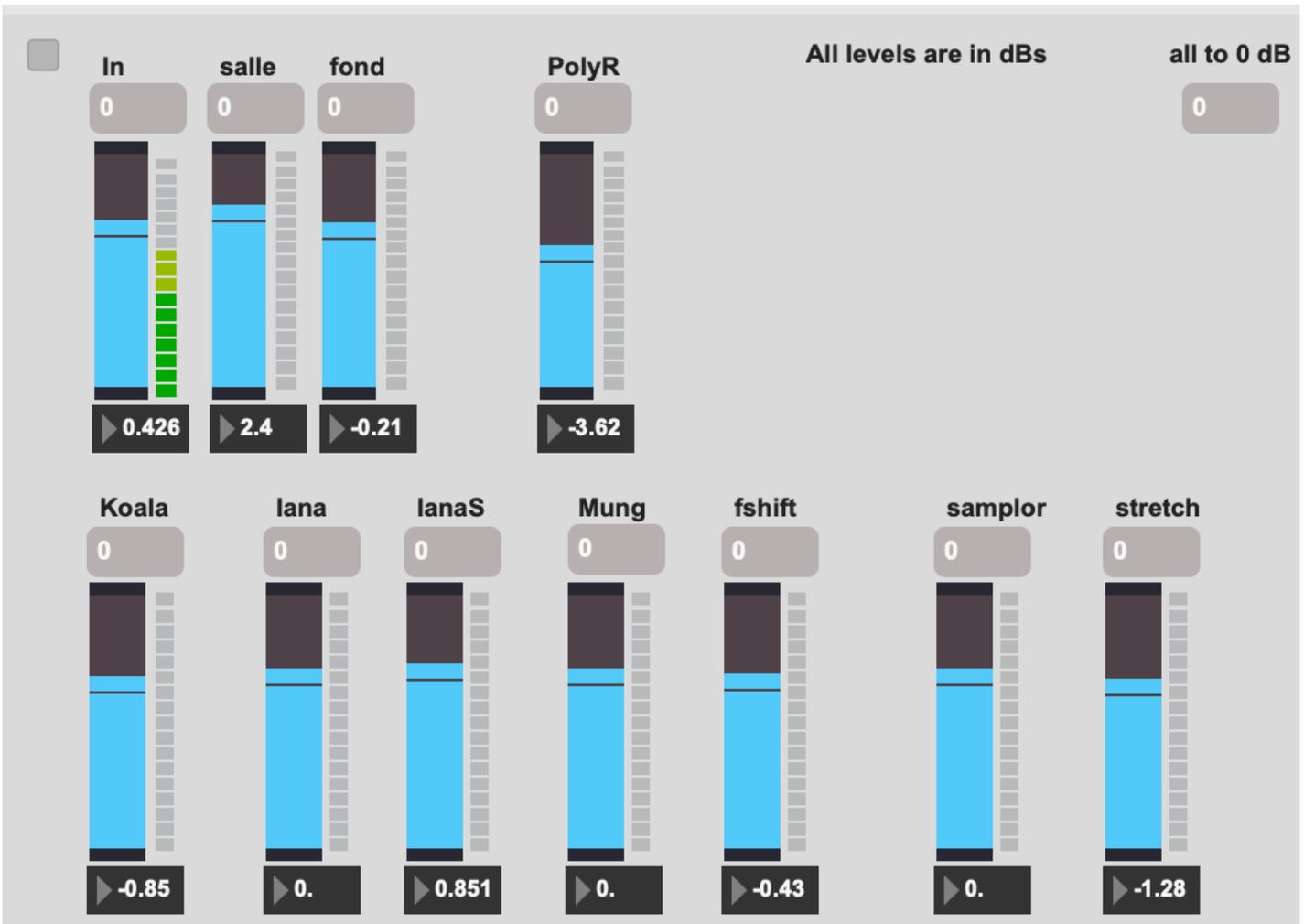
- Top Control Panel:** Includes a 'Current Event' dropdown (5) SCRUB, a 'wait....' display, and a 'Next Event' dropdown (Next: the_end). It also features a 'Launch Event' section with 'next_label' and 'repeat' buttons, and a 'CPU USAGE' monitor showing 12% current usage and a maximum of 63%.
- Antescofo Section:** Contains 'Antescofo:' controls with 'reloadscore' and 'Suivi On' buttons, and 'Antescofo message' dropdown. It also has 'start', 'stop', 'play', and 'playFrom' buttons, and 'score Tempo' and 'RT Tempo' controls.
- Right Panel:** Features a 'DSP' button, a 'RESET' button, and a 'MIDI A6' control. It also includes a '7) KEYBOARD Shortcut >>>' control and a 'Matrice Interface' section with 'Mix window front' and 'SPAT OPER' controls.
- Effect Modules Grid:** A large grid of 30 effect modules, each with a level meter and gain control. The modules are: 1DAC, 2DAC, 3DAC, 1Ryt, 1spat, 1bus, 2bus, 1revg, 1cas, 2cas, 3cas, 4cas, 5cas, 1psy, 2psy, 3psy, 4psy, 1Mng, 1cho, 1Chr, 1comp, 1Yis, 1Biq, 1Fsh, 1Giz, 1gran, 1Hrm, 1iana, 2iana, 3iana, 1ians, 2ians, 3ians, 4ians, 5ians, 6ians.

Effects are :

- **1Ryt** > Virtual instance to communicate with PRTM
- **1spat** > Ircam Spatialisateur
- **1bus...2bus** > Virtual Bus
- **1Biq** > Biquad filter
- **1revg** > Reverb
- **1cas...6cas** > Cascade Filters
- **1cho** > Chorus effect
- **1Chr** > Spectral Delay with Chromax® algorithm
- **1comp** > compressor
- **1FKo...2FKo** > spectral spread effect
- **1Fsh...2Fsh** > Frequency Shifting
- **1Giz** > Gizmo transposition
- **1gran** > Granulator
- **1Hrm** > Harmonizer
- **1iana...3iana** > Iana analysis and resynthesis
- **1ans...6ians** > sinusoidal synthesis working with 1/2/3iana
- **1Mng** > Granulation (Munger®)
- **1psy...4psy** > Psychoirtrist®
- **1samp...4samp** > sampler
- **1str...2str** > Stretch with SuperVP®
- **1Yis** > Yin analysis to sinusoidal synthesis

MiraWeb :

A Mira web interface is displayed for control and mixing on the right of the main patch. <https://cycling74.com/products/mira>



PTRM Patch :

Controlled by the main patch and Antescofo

Antescofo :

Antescofo is used to manage and recall effects preset, like an electronic score. Open `_/Tutti-Main/PATCH_TUTTI_2016_MAX7/ANTESCOFO/_Tutti.asco.txt` in a text editor to see or modify the score. All events are triggered by MIDI keyboard.

To start directly to an event, click RESET, and select the event you want to jump to. Antescofo will wait a "next_event" or the corresponding note from MIDI keyboard to launch the wanted event.

(FYI, “Jump to” launch all events from the beginning of the score to the previous event you choose.)

Click button “repeat” to repeat the same event.

Click button “next_label” to go to the next event (or to trigger event chosen with umenu “Go to Cues”)

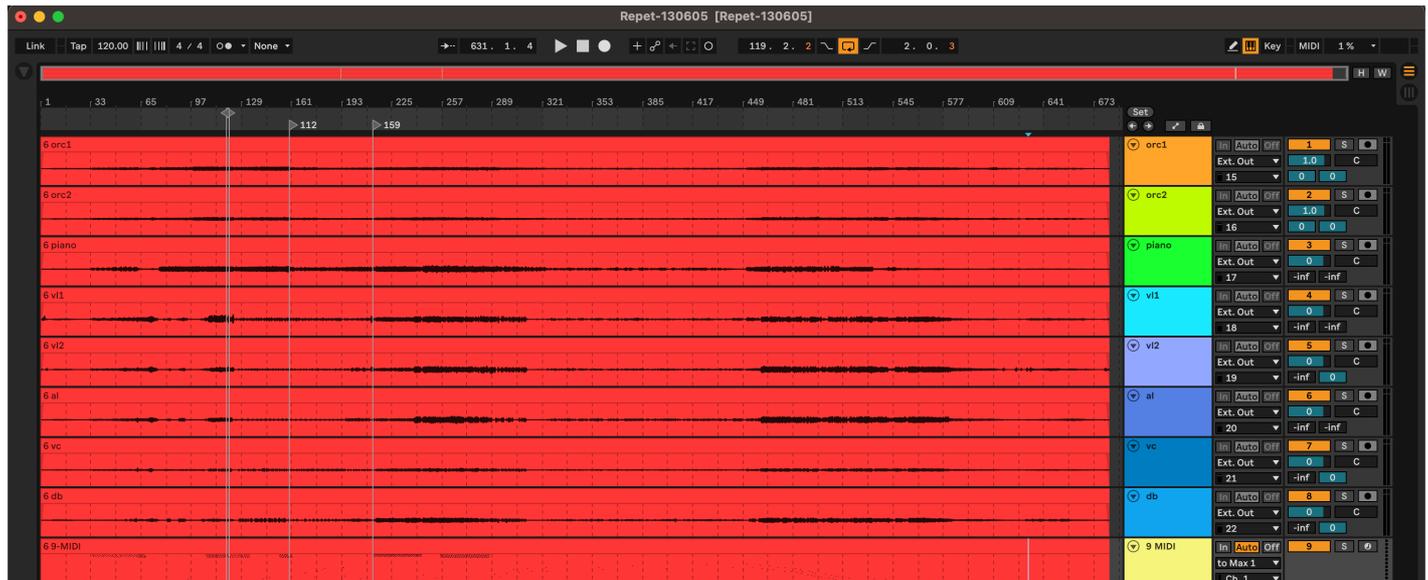
Click red button “RESET” to initialize all the patch.

Performance Notes

You will found some notes written in /notes.txt

Simulation

A multitrack recording of a rehearsal with midi trigger is at your disposal in an Ableton Live Session



© IRCAM  BY-NC-ND

This documentation is licensed under a [Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License](https://creativecommons.org/licenses/by-nc-nd/4.0/).

Program note

Tutti est la première œuvre pour grand ensemble et électronique de Yan Maresz. Partisan d'une approche inventive et toujours renouvelée de l'outil informatique, le compositeur en profite pour prendre le contrepied de ce qui se fait à l'accoutumée dans le genre : on ne trouvera donc ici quasiment aucun traitement en temps réel — le son de l'ensemble est capté et rediffusé dans la salle, avec divers procédés d'analyse/synthèse, mais non traité —, et l'on pourra entendre une écriture de l'ensemble instrumental contrainte dans le but de laisser un véritable espace d'expression dédié à l'électronique.

« J'ai souvent le sentiment que les œuvres pour ensemble et électronique sont avant tout des œuvres pour ensemble, avec une surcouche d'électronique qui vient surcolorer une partition déjà très écrite, dit Yan Maresz. Ici, j'ai volontairement restreint l'écriture instrumentale de détail : aucun mode de jeu, aucune fioriture, rien du discours rhétorique instrumental habituel. C'est un matériau brut. Il est selon moi inutile de colorer chaque note d'un timbre spécifique par un jeu instrumental singulier, si cette couleur sera de toute façon voilée ensuite par l'électronique. Écrire pour grand ensemble et électronique exige au contraire de réinventer l'interaction entre l'espace acoustique et l'espace sonore généré par les haut-parleurs.

« Mon idée de départ a donc été d'écrire l'ensemble comme un seul et unique instrument — un seul instrument capable de produire des couleurs et un discours variés, mais porteur d'une seule identité, comme un violon ou un piano seul. Il n'y a donc que très peu de polyphonie au sein de l'ensemble : le sens du discours vient de l'articulation et de l'intégration plus étroite de l'ensemble et de l'électronique. »

Pour l'écriture de la partie électronique, Yan Maresz s'est quasiment exclusivement servi d'un logiciel d'analyse et de synthèse sur modèle rythmique dont la mise au point l'occupe depuis une quinzaine d'années. Le principe de base est d'une simplicité désarmante puisqu'elle s'appuie littéralement sur la nature ondulatoire du son, et sur l'une des méthodes d'analyse du signal les plus anciennes qui soient : l'analyse spectrale, encore appelée transformée de Fourier.

Revenons un instant sur cette « transformée » : la théorie mathématique nous affirme que tout signal peut se décomposer en une somme de courbes sinusoïdales simples, d'amplitude variées. La transformée de Fourier nous indique, pour un signal donné, quelles sont les fréquences de ces courbes sinusoïdales et leurs amplitudes associées. Mais qu'est-ce qu'une fréquence sinon le nombre d'oscillations du signal dans l'espace d'une seconde ? Résultat, à chaque fréquence correspond un tempo, un rythme. Si l'on prend donc toutes les fréquences qui composent un son quelconque donné, on peut obtenir une polyphonie de tempos, d'où découle directement un rythme, une polyrythmie — la polyrythmie étant l'une des grandes préoccupations de Maresz depuis ses débuts, on comprendra ce qui, dans cette méthode, attire le compositeur. Certes, ces tempi sont en l'espèce souvent beaucoup trop rapides pour être perçus, mais rien ne nous empêche de les multiplier tous par un même facteur, pour obtenir un véritable tissu polyrythmique audible.

On obtient ainsi un modèle rythmique, directement déduit du son que l'ordinateur a pu capter, en temps réel, de l'ensemble instrumental, ou d'un échantillon préenregistré qu'on lui aura donné à analyser.

Inversement, d'un modèle rythmique donné, on peut déduire... un accord ! En captant les différents impacts des attaques de note de l'ensemble instrumental, ce moteur peut générer un son de synthèse, un véritable timbre — dont la texture variera au cours du temps en fonction des variations rythmiques qui anime l'ensemble instrumental capté.

Bien que rudimentaire, le principe de base de ce logiciel permet une infinité de variations : en utilisant les résultats de l'analyse pour générer à la fois des fréquences et des rythmes, on peut obtenir un discours polyphonique ; au moyen de filtres fréquentiels, on peut ne retenir que certaines fréquences (ergo : certains rythmes) ; au moyen d'un pivot en fréquence au milieu du spectre, on peut, par une simple similitude, élargir ou réduire le modèle rythmique ou l'accord (selon que l'on veut générer l'un ou l'autre) ; en arrondissant les enveloppes et en jouant sur la résonance, l'ordinateur peut composer en temps réel une véritable mélodie... Dans les mains de son compositeur/concepteur, qui en explore avec enthousiasme toutes les subtilités, l'outil s'avère extrêmement souple et puissant.

« Je choisis tous les paramètres en fonction de l'état musical que je veux obtenir à un instant donné : l'ensemble de ces paramètres fixe un « état » de la machine, laquelle suit l'analyse en temps réel — ça prend à peu près 5 ms, c'est donc quasi instantané —, et produit un son vivant, une texture animée de l'intérieur, presque palpante. Les états de la machine sont si variés et différents qu'elle ne sonne jamais deux fois de la même manière. Certains moments, l'électronique va sembler très kitsch, avec des sinus qui rappellent les Ondes Martenot, ou comme des mélodies, très vieilles, très premier degré — il faut aussi savoir rire de sa musique. C'est pourtant toujours la même machine. »

L'ensemble instrumental et l'électronique jouent donc ensemble, la seconde prenant sa source dans le premier, dans un dialogue quasi concertant.

« Dans un premier temps, j'ai voulu intituler la pièce *Ripieno*, se souvient Yan Maresz : justement parce l'électronique joue souvent le rôle du solo du concerto grosso baroque. Fondue dans la masse, elle s'en détache par moments, comme une émanation familière, pour rentrer ensuite dans le rang. « Concerto », c'est « jouer ensemble », et « Ripieno », « toujours ensemble » : mis à part une présence importante du piano vers le début de la pièce, il n'y a aucun solo de toute la durée de l'œuvre. J'ai finalement préféré le terme « Tutti », tout simplement parce que le sens premier de « Ripieno » est « fourré » en italien, ce qui n'est pas très joli. »

La dynamique du concerto grosso baroque se retrouve au reste dans la spatialisation sonore. « Pendant les sections que l'on pourrait assimiler au « Tutti » du concerto grosso, l'électronique est diffusée au même niveau que l'ensemble, par les haut-parleurs sur la scène. Lorsqu'elle commence à s'autonomiser, à sortir de la masse orchestrale, quand son discours s'affranchit de celui de l'ensemble, elle semble alors en sortir véritablement : la diffusion sonore ne sera plus sur scène mais dans la salle, laissant l'ensemble dans l'arrière plan. La texture même des sons ressemblera plus spécifiquement à de la musique électroacoustique. Se dégagera ainsi une forme dramaturgique

spatiale, qui donne un supplément de vue au discours électronique. »

Lorsqu'on aborde la question de la poétique de la pièce, toutefois, Yan Maresz cherche ses mots : « Je suis incapable d'anticiper l'impact psychologique de ma musique au moment de l'écriture : je ne découvre sa poétique qu'en l'écoutant.

« Je travaille avec des idées musicales, j'entends des atmosphères, des références, des doigts tendus vers des lieux de mémoire, des commentaires. Comme un éclairage sur une expérience partagée dans le passé : car je ne me considère pas comme un homme visionnaire. Je côtoie constamment les compositeurs les plus prospectifs, mais je ne crois plus vraiment à un quelconque discours sur la modernité.

« Mon sentiment est que le rythme, la pulsation, la période, tout ce qui constitue les fondamentaux de l'écriture ne sont jamais qu'une capture d'un infini — comme le modèle théorique du sinus, qui ne démarre ni ne finit jamais. C'est ainsi que s'ouvre *Tutti*, par la capture d'un moment glacé : le premier son qu'on entendra sera un accord d'orchestre de vingt-cinq secondes, complètement immobile, point de départ de l'électronique et de tout le discours musical.

« La poétique naît toute seule, conclut Yan Maresz. Je me souviens de la réponse qu'a un jour donnée [Luciano Berio](#) à la question : « Qu'est-ce que la musique ? » Après un laïus d'une cinquantaine de pages, il concluait par : « Personne n'aurait eu l'idée de poser cette question à Beethoven ! » La musique, c'est ce qu'on a envie d'écouter avec l'intention d'écouter de la musique. Point. »

Jérémie Szpirglas, festival ManiFeste, 6 juin 2013.

Version documentation creation date: April 15, 2024, 11:34 a.m., update date: May 20, 2024, 1:49 p.m.