



HAL
open science

ML-Maquette / Musique Lab 2

Jean Bresson

► **To cite this version:**

Jean Bresson. ML-Maquette / Musique Lab 2. International Computer Music Conference (ICMC'10), 2010, New York / Stony Brook, United States. hal-01161248

HAL Id: hal-01161248

<https://hal.science/hal-01161248>

Submitted on 8 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ML-MAQUETTE / MUSIQUE LAB 2

Jean Bresson

IRCAM – CNRS UMR STMS

Paris, France

bresson@ircam.fr

ABSTRACT

Musique Lab 2 is an educational application providing music teachers with means to develop new pedagogical approaches and situations using computer tools. It is based and developed on top of the OpenMusic computer-aided composition environment and benefits from the underlying music representation, computational possibilities and from the extended compositional knowledge and practice existing in this environment.

1. INTRODUCTION

The Musique Lab project is the object of a collaboration between IRCAM and the French Ministry of Education. After the release, in 2002, of a first software suite introducing basic musical concepts such as pitches, scales, rhythms or dynamics, a second phase in this project, Musique Lab 2 (ML2), started in 2004 with new objectives related to the integration of advanced tools and formalisms inspired from IRCAM musical research. In this context, three prototype applications have been created: ML-Audio, ML-Annotation, and ML-Maquette [12, 8]. Eventually, the project concentrated on ML-Maquette, which has been renamed and distributed as “Musique Lab 2”.

ML-Maquette has been designed as a pedagogical application built on top of OpenMusic (OM), a computer-aided composition environment also created at IRCAM [1, 2]. OM is a visual programming environment used by composers to create processes for the generation and manipulation of musical structures. As such it provides both expressive and computational power and allows musicians to design wide varieties of programs corresponding to their particular needs and musical approaches. A considerable set of musical tools have been developed by composers and researchers in this environment, most of which are made available to the users in the OM distribution or through external libraries. OM also provides innovative time representations, particularly with the *maquette*, a programmable sequencing interface allowing to organize programs and data in a temporal structure.

In order to benefit from the musical representations and computational features of OM without the programming aspects, which would have been out of the scope in the context of music teaching, a single all-in-one interface has been designed for this project. This interface (see Figure 1) represents a pedagogical document corresponding to a specific teaching situation to be prepared by the teacher or developed and manipulated with his/her students during a music lesson.

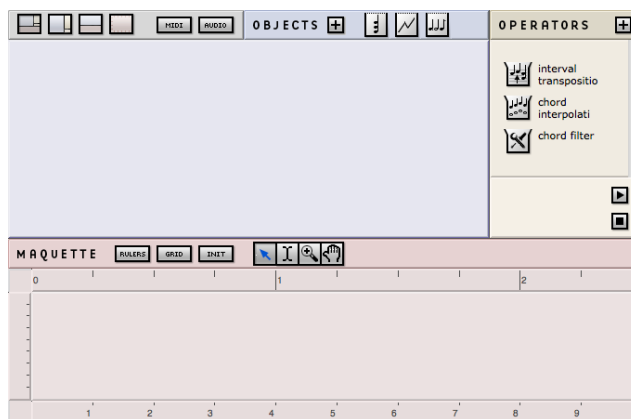


Figure 1. Musique Lab 2 (ML-Maquette) main window.

2. BASICS AND USER INTERACTION

A document is made of 3 areas, called respectively “objects”, “operators”, and “maquette”. Most of the interaction is based on drag-and-drop operations: The successive steps in the “life” of musical objects (instantiation, construction, transformation, organization) are triggered by dragging icons on the different document areas.

The “objects” area (left-hand part of the window) allows to instantiate and store musical objects selected from a small set of available classes: chords, sequences, rhythmic sequences, envelopes (curves or break-point functions—BPF) or sounds (audio files). These objects are represented by boxes (see Figure 2) and can be opened and edited using specific editors (score editors, BPF editor, etc.)

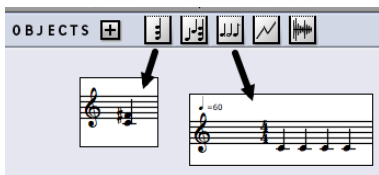


Figure 2. Instantiating objects.

The “operators” area (right-hand part of the window) contains a list of operators chosen by the user while creating his/her document. These operators perform transformations on the objects and generate new ones. By dropping an object on one of these operators, and after specifying some additional parameters, a new object is created and appears in the bottom part of the operators area (see Figure 3 with the example of the *arpeggio* operator). This new object can then be dropped and used among the other ones in the “objects” area.

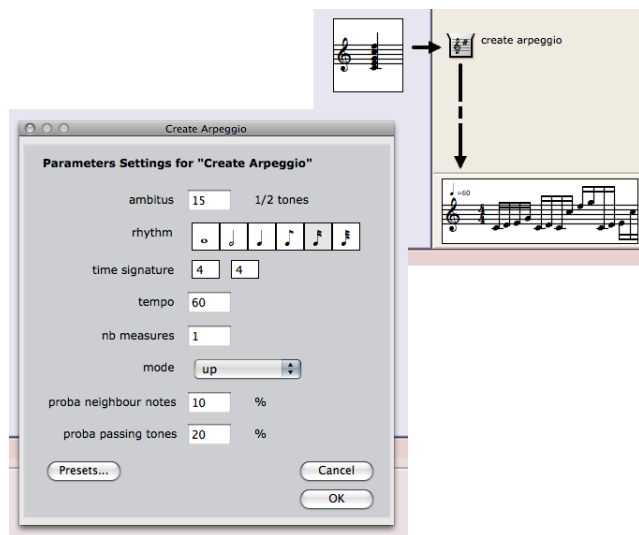


Figure 3. Application of an operator. A dialog window appears when the initial object is dropped on the operator icon, and allows to set the operator’s parameters.

Finally, the “maquette” area (bottom part of the window) is a sequencer allowing to organize the objects previously created in a common temporal referential. This area has also powerful functional features: It is possible to set a specific behaviour to the boxes it contains, so that their position and/or dimensions dynamically modify some of their attributes, or even recompute them by changing some parameters of the operator from which they were created.

Figure 4 shows an object created from the *arpeggio* operator (see Figure 3), dropped and duplicated in the maquette area. The box settings have been parametrized so that its vertical position would change the range (“ambitus”) of the arpeggio.

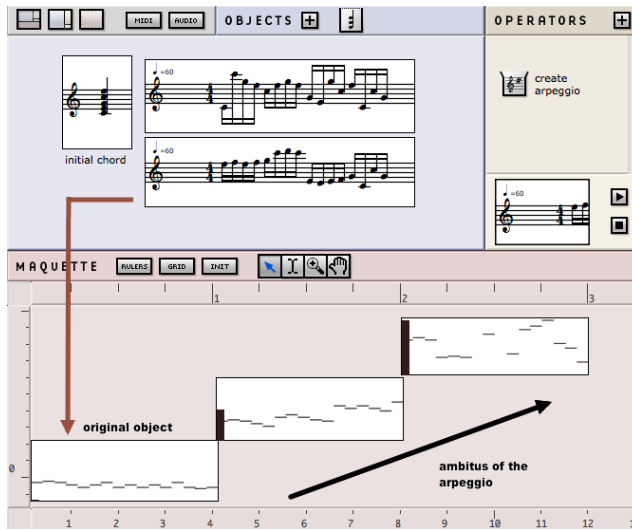


Figure 4. The maquette area: objects can be recomputed depending of their size and position.

3. OPERATORS AND APPLICATIONS

The operators available in the application library are related to varied musical domains and organized in different categories. These categories begin with general-purpose manipulations on musical structures and then get specialized to specific musical or extra-musical domains such as tonality, counterpoint, curve manipulations, or signal processing.

The more general tools include concatenations, superimposition, rhythmic quantification, and other utilities generally coming from commonly used OM functions and processes. The envelope-processing operators are also derived from existing OM tools and perform rescaling, re-sampling, smoothing and other processes on envelopes, be they curves (BPFs) or melodic profiles (see Figure 5).

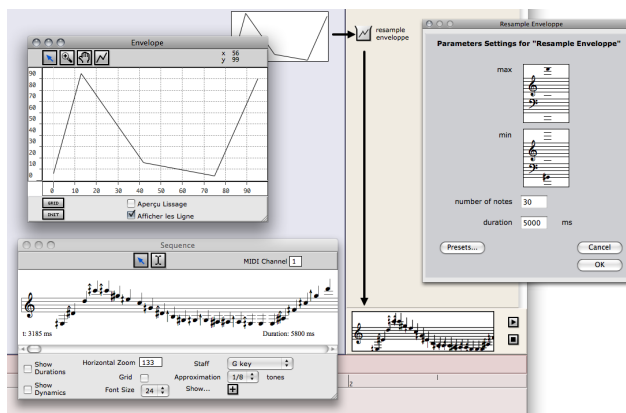


Figure 5. Using envelope processing operators: generation of note sequences from hand-drawn curves and resampling algorithms.

A group of operators is inspired from the OM *Profile* library functions and performs several types of interpolations between chords or melodies (see Figure 6).

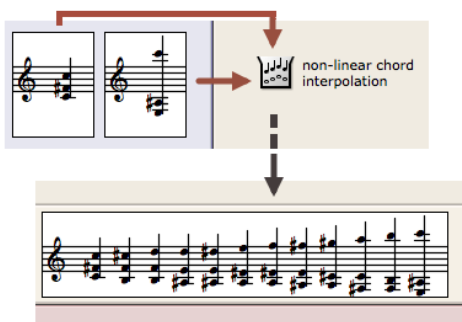


Figure 6. Chord interpolation. The first chord is dropped on the operator, and the second one is to be dropped in the parameters window of the operator (not visible on the figure).

Another part of the operators has been designed specifically for the pedagogical purpose of the Musique Lab project, a number of them working in the tonal domain.

A hierarchical model of tonality has been implemented and integrated in the OM musical objects: Each object can have a tonality, and its internal components (e.g. the chords in a sequence) either have their own, different tonality (which constitutes a kind of modulation), or inherit the tonality from their container. Tonal issues can therefore be considered in the representation of the musical objects (mostly, in the graphical editors) and in their internal structure during computations, which makes it possible to perform tonal transpositions and modulations, counterpoint operations, generate arpeggios, harmonic steps and the like. Figure 7 is an example of counterpoint operations, applied considering the tonality of an original sequence.

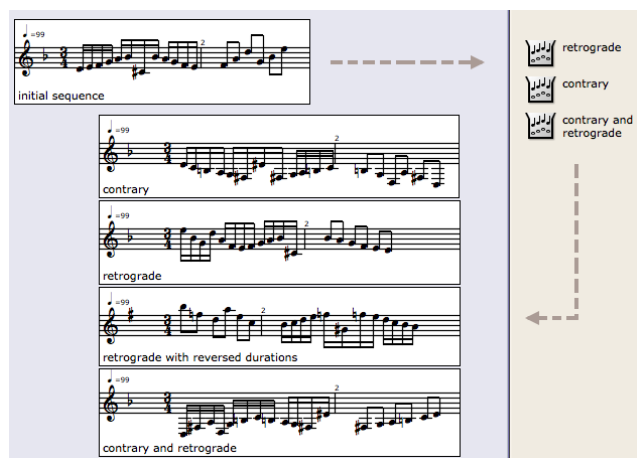


Figure 7. Example of tonal operators: retrograde and contrary movements.

Finally, a group of signal processing operators perform sound file analyses such as the fundamental frequency estimation [6] of the signal, or the partial tracking [11] and “chord-sequence” analyses, whose results are converted to symbolic chord sequences [9] (see Figure 8). They are also completed with an additive synthesis tool allowing to create sounds out of symbolic musical objects. All these operators are based on pm2, an additive sound analysis and synthesis program which is notably used as one of the underlying analysis kernel in the IRCAM AudioSculpt software [3]. An outstanding character of these tools is the relative simplicity with which the previous operations can be performed and linked to more general musical contexts or pedagogical approaches, which makes fundamental contemporary music issues such as the relations between sound and symbolic levels available and easily accessible to any music student or teacher.

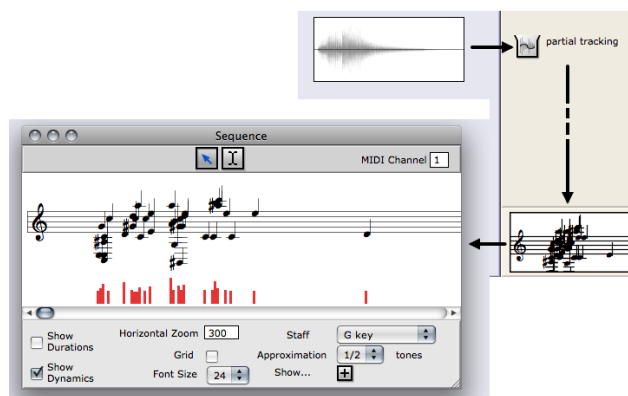


Figure 8. A sound processing operator: Partial tracking.

The set operators available in Musique Lab 2 is of course extensible, and will probably be completed in the future. A relatively simple protocol allows to integrate new operators programmed as Lisp functions or coming from OM (remember that the OM kernel is running under the application). Once a new operator is declared, all the graphical components (and particularly the parameters setting dialogs and interface) are set up and displayed automatically. This behaviour, principally implemented thanks to the method inspection features provided by OM and the CLOS metaobject protocol [7, 10], is described with more details in [4].

4. CONCLUSION

A description of ML-Maquette / Musique Lab 2 can be found in [5], and additional pedagogical issues are given in [8] (references in French).

The application currently runs on Windows and MacOSX operating system. It is distributed via the IRCAM forum¹ and on the internal French Ministry of Education network.

¹<http://forumnet.ircam.fr/>

A set of documented examples and documents will be released with the software, as a starting point for users to develop their own teaching supports or experiences. Figure 9 shows one of these documents, centered on Philippe Leroux's piece *Voi(REX)*.

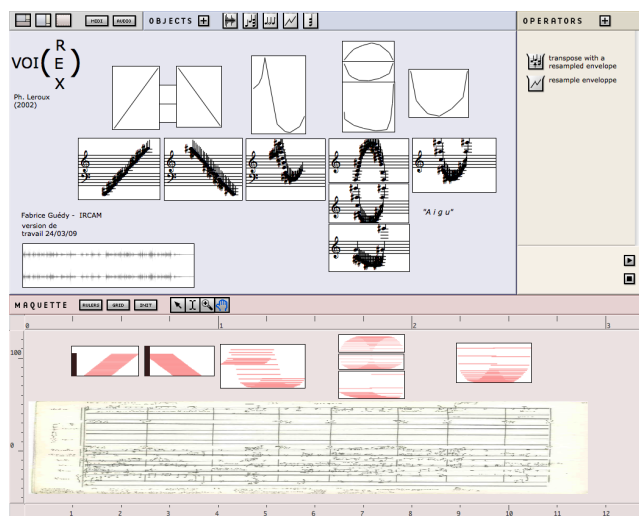


Figure 9. Reconstitution and pedagogical approach of Philippe Leroux's piece *Voi(REX)* in Musique Lab 2 (by F. Guédy).

5. ACKNOWLEDGEMENTS

The Musique Lab 2 project benefits from financial and pedagogical support from the French Ministry of Education. Software conception and development by the author at IRCAM, Music Representation team with contributions by Gérard Assayag, Carlos Agon and Charlotte Truchet. Audio/MIDI support: LibAudioStream/MidiShare by Grame. Project coordination by Cyril Béros, IRCAM Education and Cultural Outreach department. Pedagogical coordination at IRCAM by Fabrice Guédy; correspondent French Ministry of Education: Matthieu Roy.

6. REFERENCES

- [1] C. Agon, "OpenMusic : Un langage visuel pour la Composition Assistée par Ordinateur," Ph.D. dissertation, Université Pierre et Marie Curie – Paris 6, France, 1998.
- [2] G. Assayag, C. Rueda, M. Laurson, C. Agon, and O. Delerue, "Computer Assisted Composition at Ir-cam: From PatchWork to OpenMusic," *Computer Music Journal*, vol. 23, no. 3, 1999.

- [3] N. Bogaards and A. Röbel, "An interface for analysis-driven sound processing," in *AES 119th Convention*, New York, USA, 2004.
- [4] J. Bresson, "Processus compositionnels et opérateurs musicaux dans ML-Maquette. Les outils de traitement du signal," in *Actes des Journées d'Informatique Musicale*, Lyon, France, 2007.
- [5] J. Bresson, F. Guédy, and G. Assayag, "Musique Lab Maquette : approche interactive des processus compositionnels pour la pédagogie musicale," *Sciences et Technologies de l'Information et de la Communication pour l'Education et la Formation*, vol. 13, 2006.
- [6] B. Doval and X. Rodet, "Estimation of Fundamental Frequency of Musical Sound Signals." in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing – ICASSP*, Toronto, Canada, 1991.
- [7] R. P. Gabriel, J. L. White, and D. G. Bobrow, "CLOS: Integrating Object-oriented and Functional Programming," *Communications of the ACM*, vol. 34, no. 9, 1991.
- [8] F. Guédy, J. Bresson, and G. Assayag, "Musique Lab 2 - Un environnement d'aide à la pédagogie musicale," in *Actes des Journées d'Informatique Musicale*, Lyon, France, 2007.
- [9] P. Hanappe and G. Assayag, "Intégration des représentations temps/fréquence et des représentations musicales symboliques," in *Recherches et applications en informatique musicale*, M. Chemiller and F. Pachet, Eds. France: Hermes, 1998.
- [10] G. Kiczales, J. des Rivières, and D. G. Bobrow, *The Art of the Metaobject Protocol*. MIT Press, 1991.
- [11] R. J. McAulay and T. F. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 4, 1986.
- [12] V. Puig, F. Guédy, M. Fingerhut, F. Serrière, J. Bresson, and O. Zeller., "Musique Lab 2: A Three-Level Approach for Music Education at School," in *Proceedings of the International Computer Music Conference*, Barcelona, Spain, 2005.