
Teach programming and composition with **OpenMusic**

Dimitri Bouche
PhD. Student @ IRCAM
Paris, France

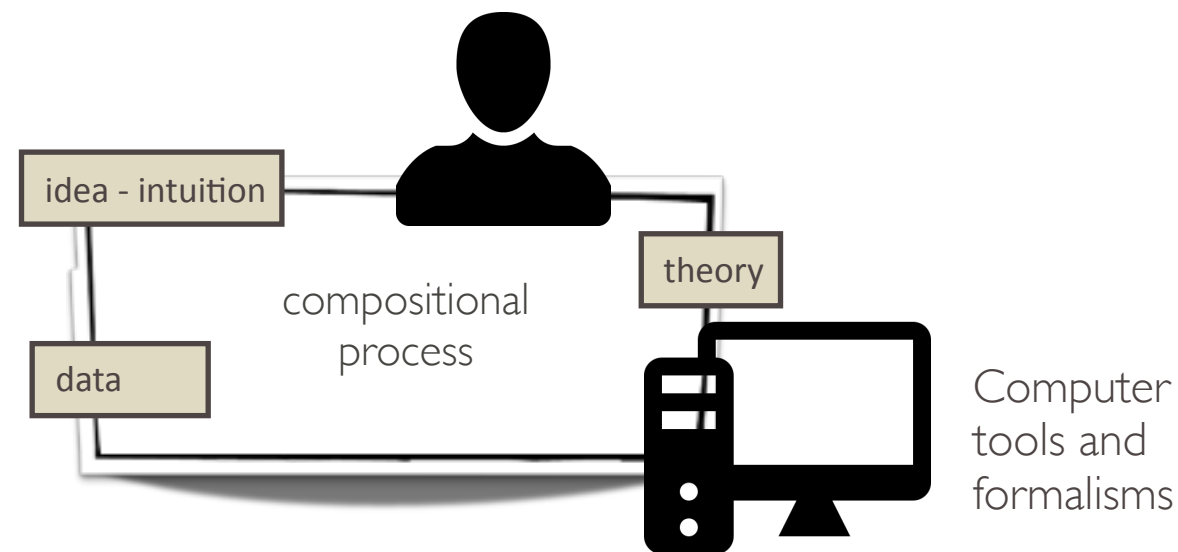
Innovative Tools and Methods to Teach Music and Signal Processing

OpenMusic introduction



Computer-Aided Composition (CAC)

The composer-programmer



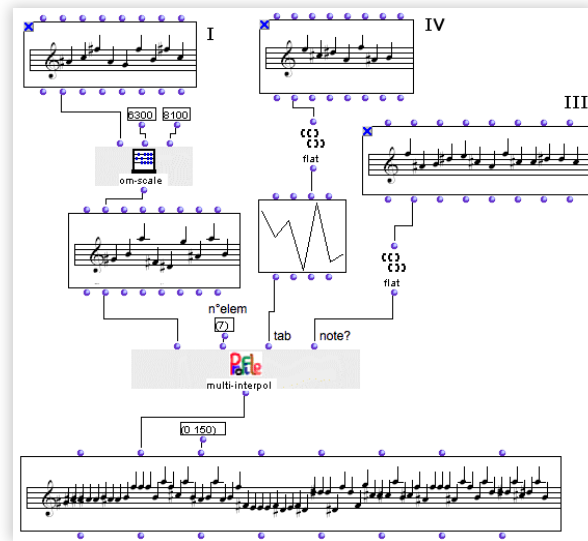
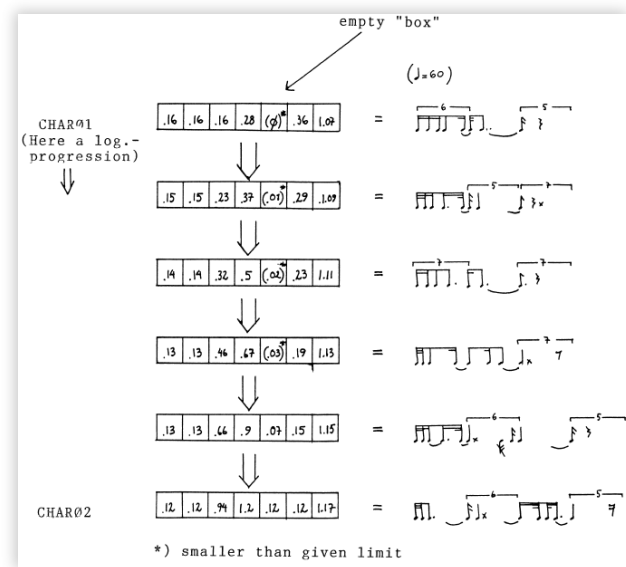
« We conceive such an environment [of computer-aided composition] as a specialized computer language that composers will use to build their own musical universe. [...] This leads us to reflect on the various existing programming models, as well as on the interfaces [...] which make it possible to control this programming, and on the representations of the musical structures, which will be built and transformed using this programming. »

G.Assayag (1998)

- G.Assayag « **Computer Assisted Composition Today** », First Symposium on Music and Computers, Corfu, 1998.

OpenMusic

Ideas + Formalization \longleftrightarrow Implementation \longleftrightarrow Score



- OM Visual program = Symbolic Representation of:
 - a musical object/process,
 - a compositional model.
- Describe intentions through a specific (computer) language based on Lisp.

<http://repmus.ircam.fr/openmusic>

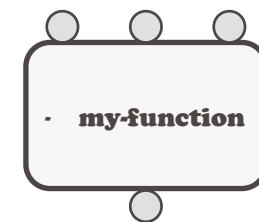
Cross-platform

OpenMusic for teaching **programming**

Visual programming

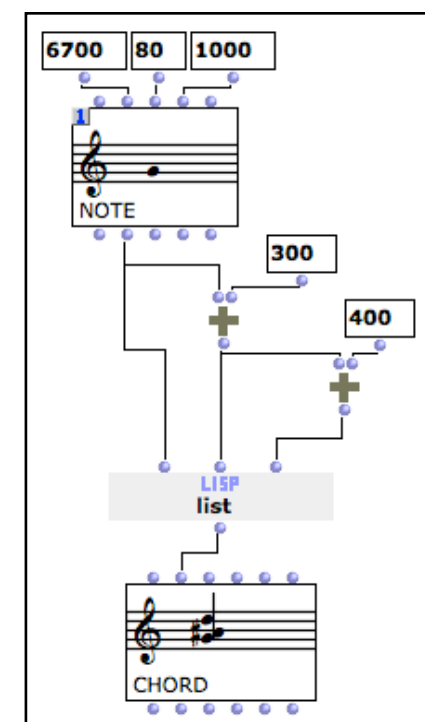
A Lisp function can be visually instantiated

```
(defun my-function (arg1 arg2 arg3)
  (let (rep)
    [...]
    rep))
```



A visual program can be converted to a Lisp function

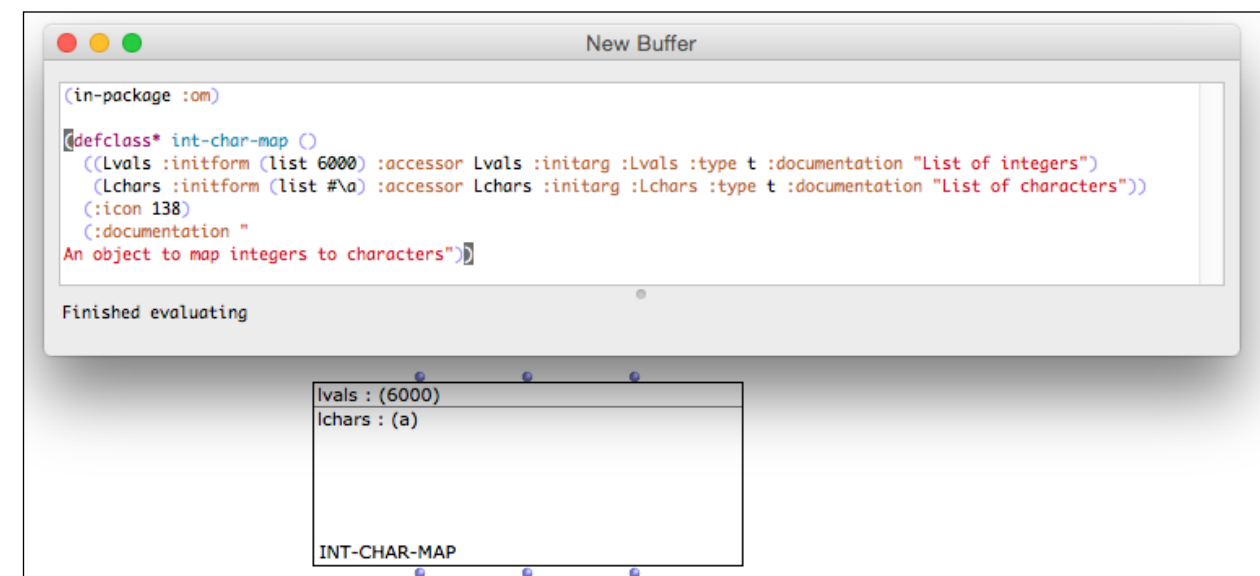
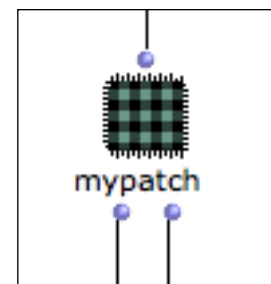
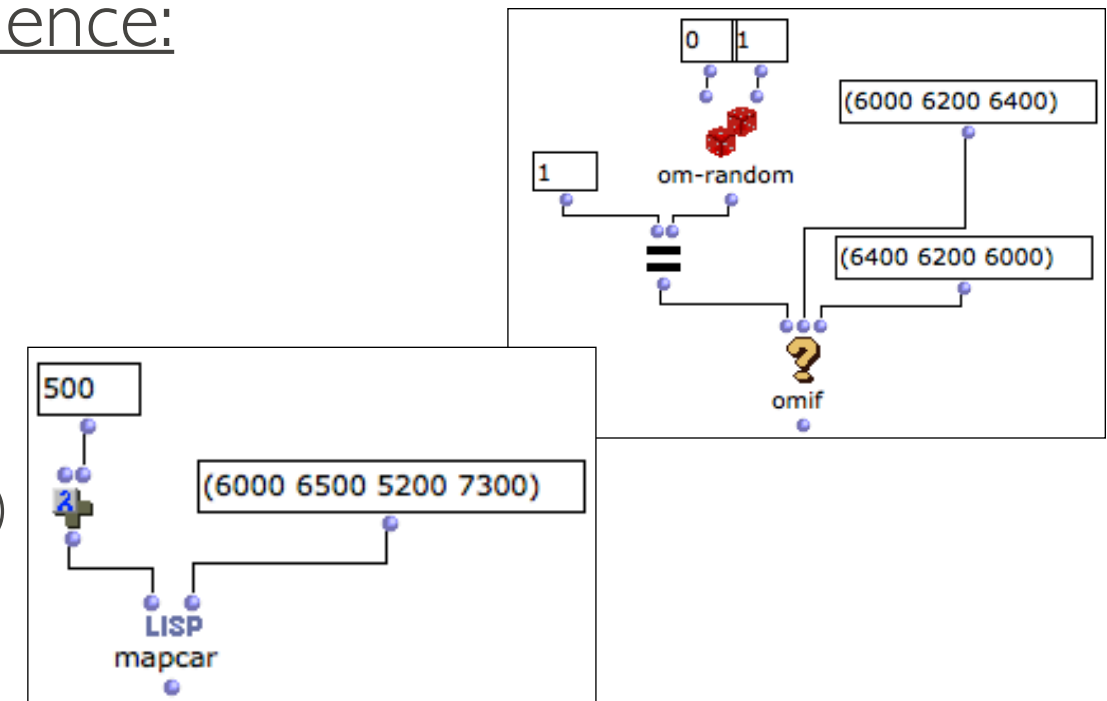
```
(let ((note (make-instance 'note
  :pitch 6700
  :vel 80
  :dur 1000)))
  (make-instance 'chord
    :pitches
    (list (pitch note)
          (+ (pitch note) 300)
          (+ (+ (pitch note) 300) 400)))))
```



Learn advanced programming without noticing

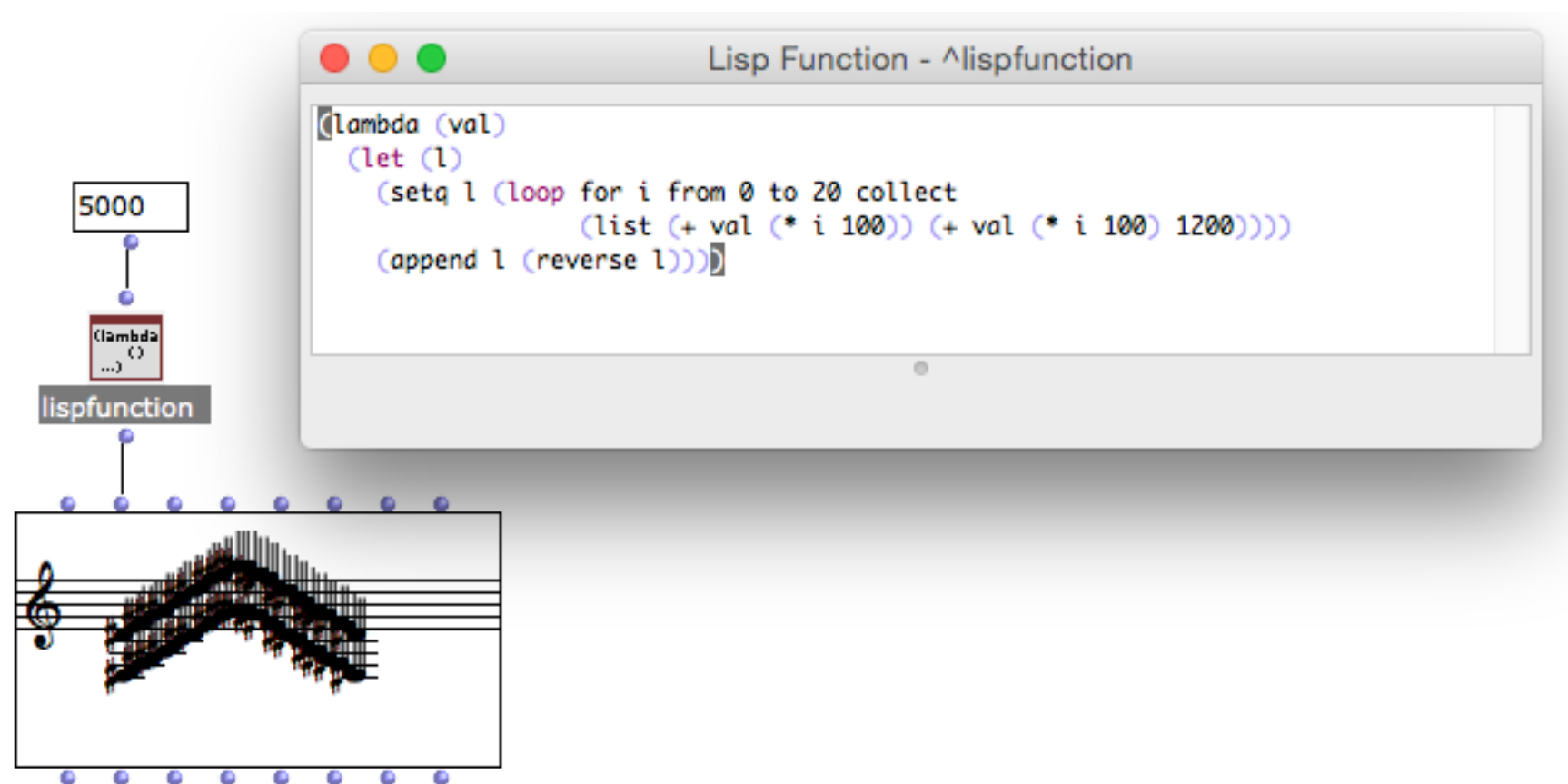
Visual programming allows to build complex programs, without being familiar with computer science:

- Conditional structures
- Lambda function (function as a parameter of an other)
- Abstraction (patch in a patch etc.)
- Object-oriented programming.



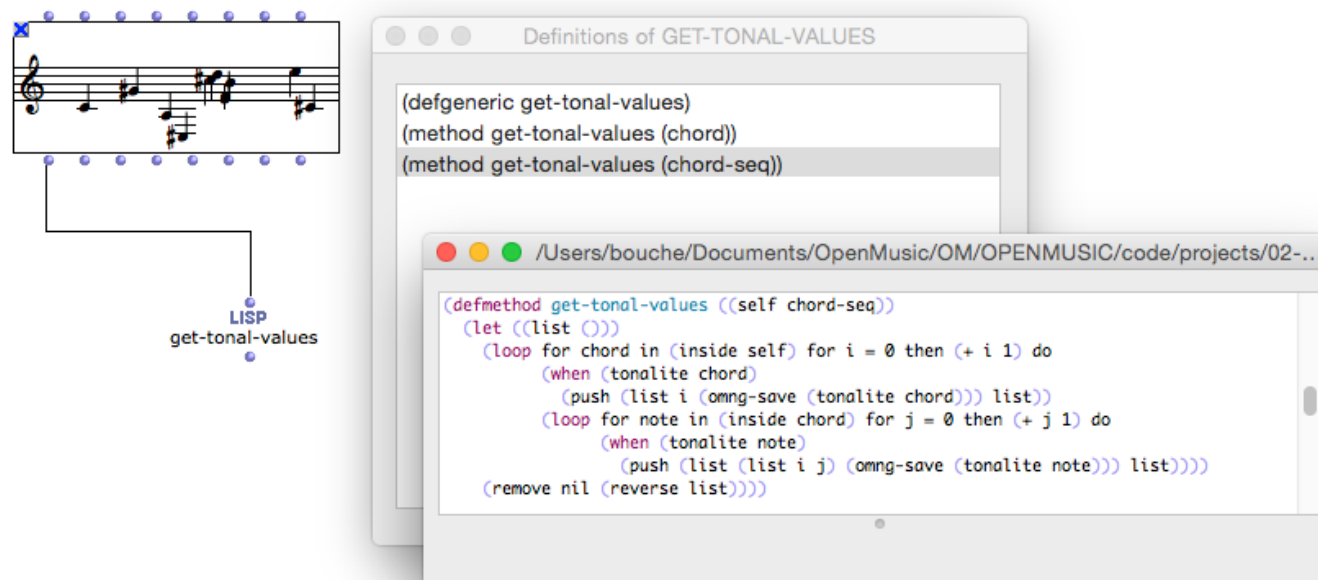
Patching with code

Lisp code can be used in a patch using the « *lispfunction* » object



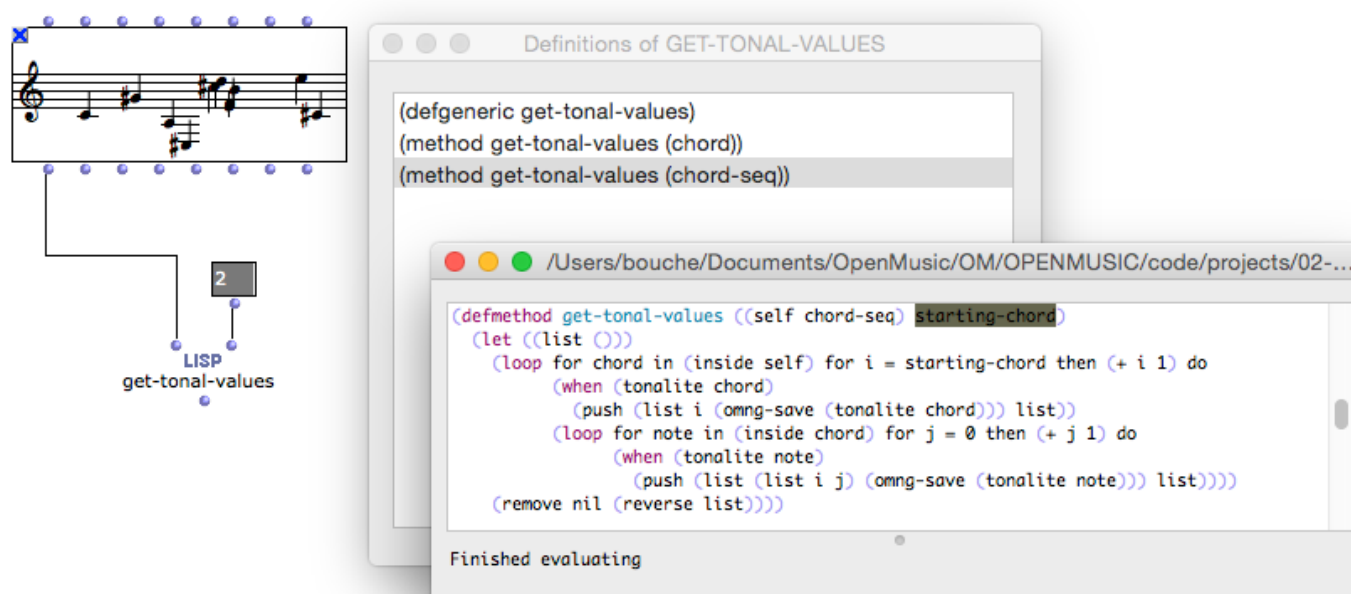
Source code

Source code of each box can be accessed...



- to understand

...it can be modified and interpreted



- to customize

OpenMusic for teaching **composition**

Teaching and learning composition

Musique Lab 2:

- Coat OpenMusic engine in a user-friendly/single-window interface
- Drag&Drop only

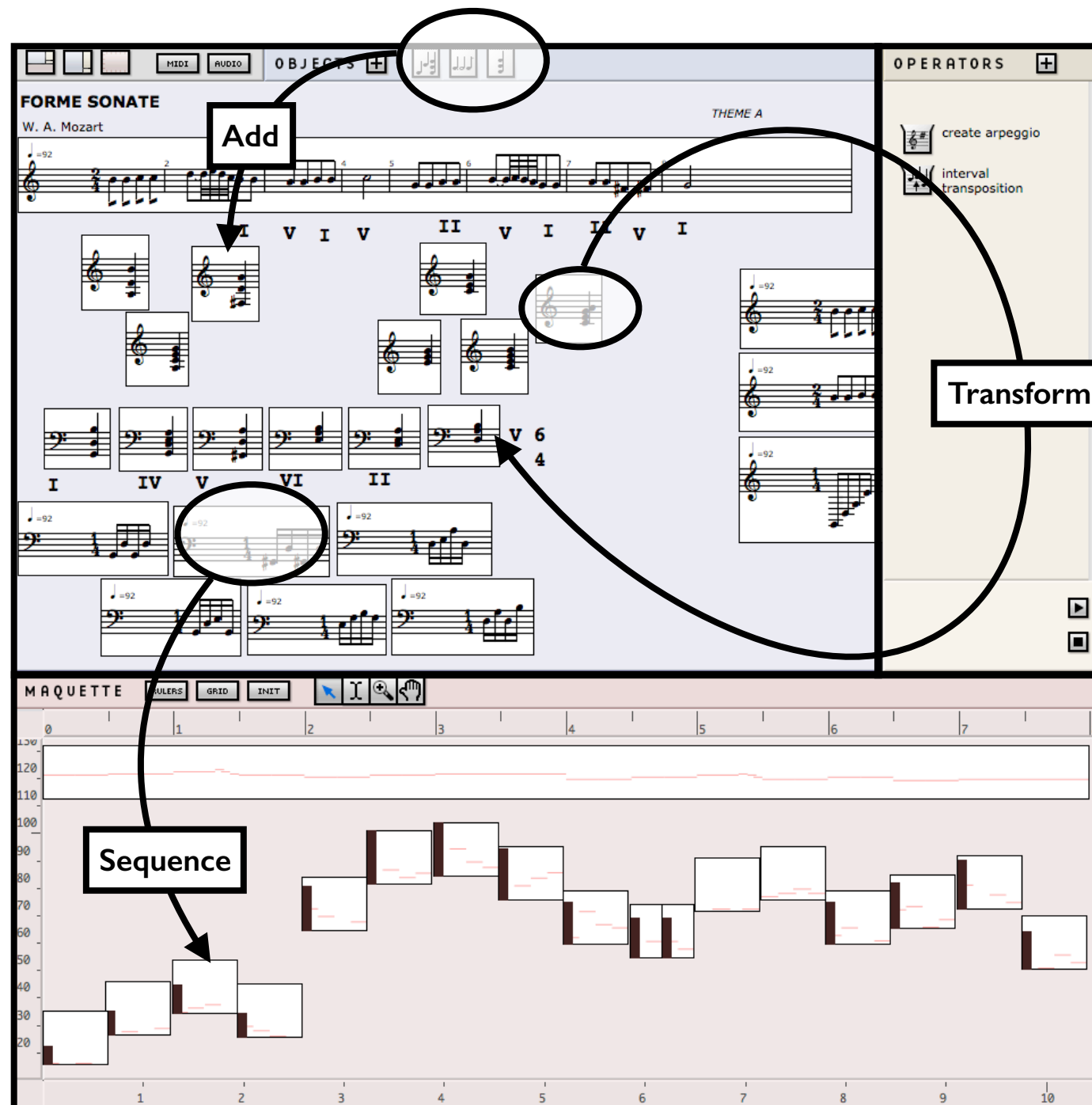
« Reactive » OpenMusic:

- From demand-driven to data-driven computation
- Propagation of editions in a patch

Scheduling model for extended CAC:

- From static to dynamic scheduling
- Rendering of structures being edited
- Processes sequencing

Teaching with Musique Lab 2



Operators (hidden processes):

- drag an object on an operator
- get the result back in the pool

Sequencer

- drag objects from the pool on a timeline
- use the y-coordinate
- each object remember its last operation

Musique Lab 2 examples

J.S. Bach
Variations Goldberg
Fabrice Guédy - IRCAM
version de
travail 24/03/09

Variation 12 : canon à la quarte

Enregistrement de référence
(Glenn Gould)

Tempo = 100

initial sequence

Basse de l'aria

sequence initiale quantifiée

MODE D'EMPLOI

- 1) transposition à la quinte (quarte inférieure)
- 2) mouvement contraire (outil "inversion")
- 3) transposition 1 degré - 1 octave
- 4) quantifier l'antécédent
- 5) mouvement contraire obtenu sur permutation d'accords
- 6) et voilà, faire glisser le résultat dans la maquettes.

TRANSPOSITION quarte inférieure

mouvement contraire

transposé de 1 degré - 1 octave

permutation sur les durées de la sequence initiale

OPERATORS

- retrograde
- contrary
- transpose on degrees
- permutation of chords on durations
- rhythmic quantification

MAQUETTE RULERS GRID INIT

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65

110 100 90 80 70 60 50 40

10 20 30 40 50 60 70 80 90 100 110

Learn counterpoint and transposition by replicating J.-S. Bach « Goldberg Variations »:

- Overview,
- Operating instructions.

tutorial by F.Guédy

Musique Lab 2 examples

VOI(R E X)
Ph. Leroux
(2002)

Fabrice Guédy - IRCAM
version de
travail 24/03/09

MAQUETTE

OPERATORS

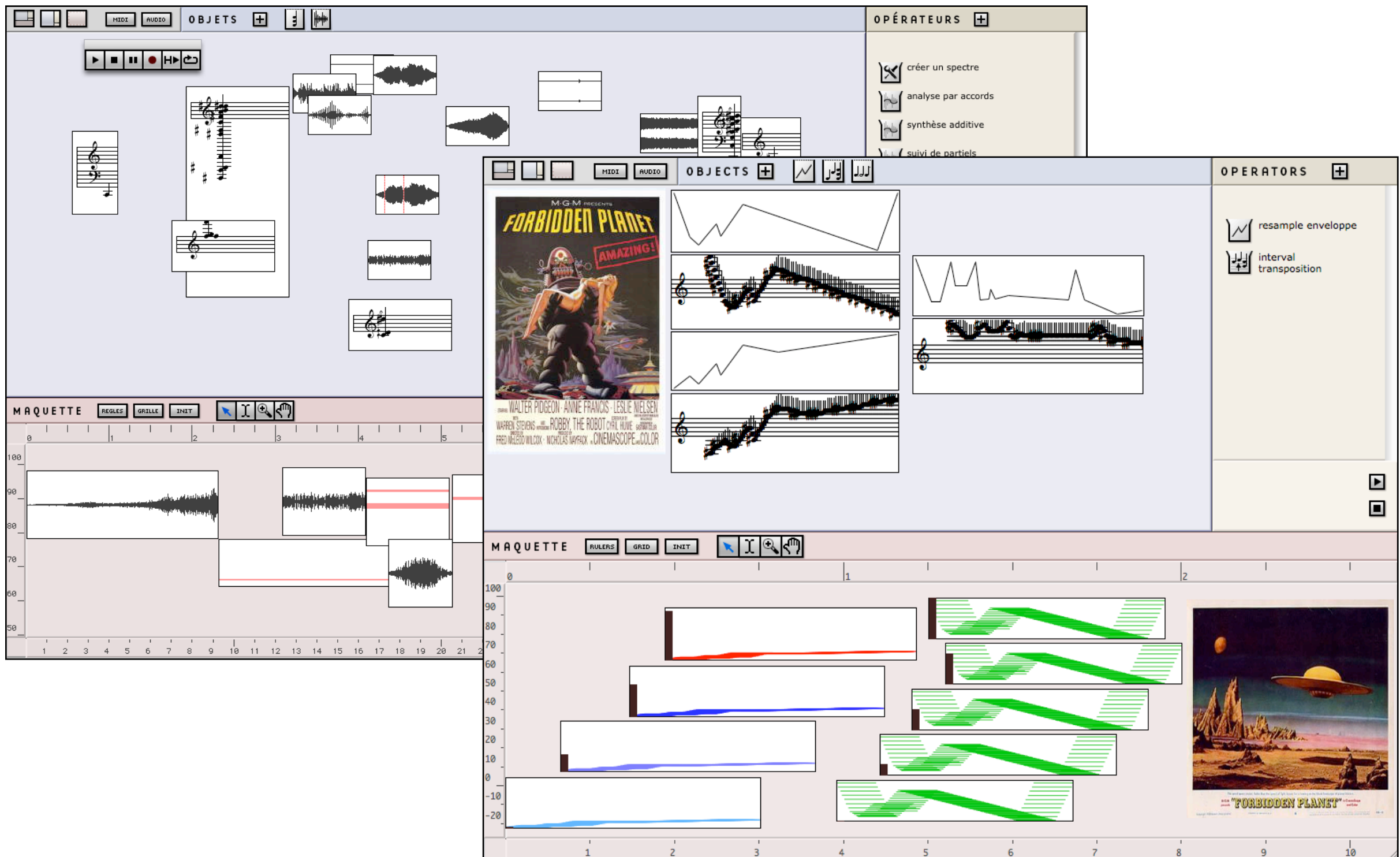
- transpose with a resampled envelope
- resample envelope

Learn using curves and graphics
in composition by replicating
P.Leroux « VOI(R E X) »:

- Overview,
- Operating instructions,
- Use the y-axis of the sequencer.

tutorial by F.Guédy

Musique Lab 2 pupil's works

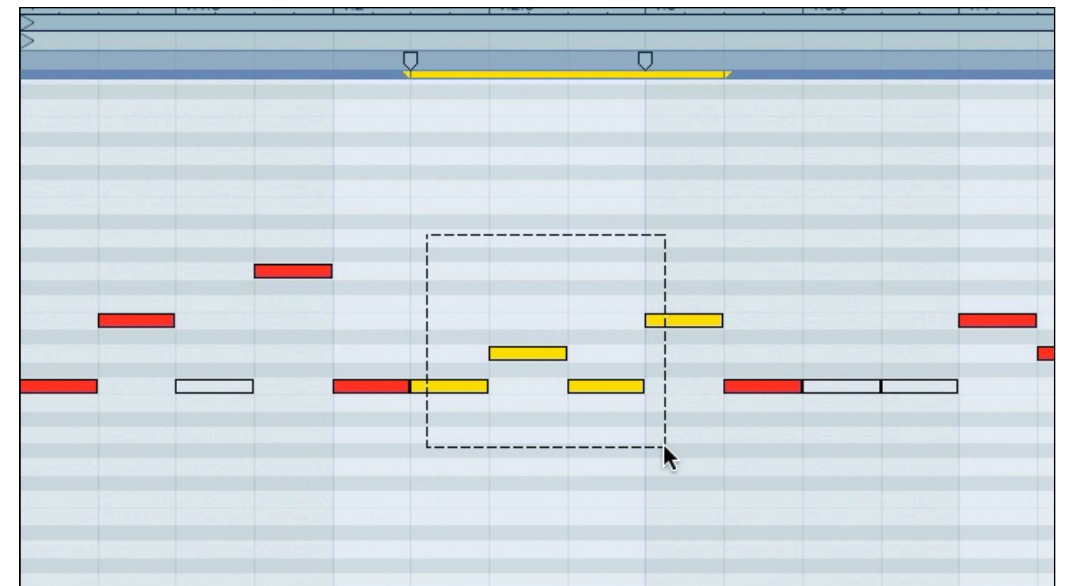
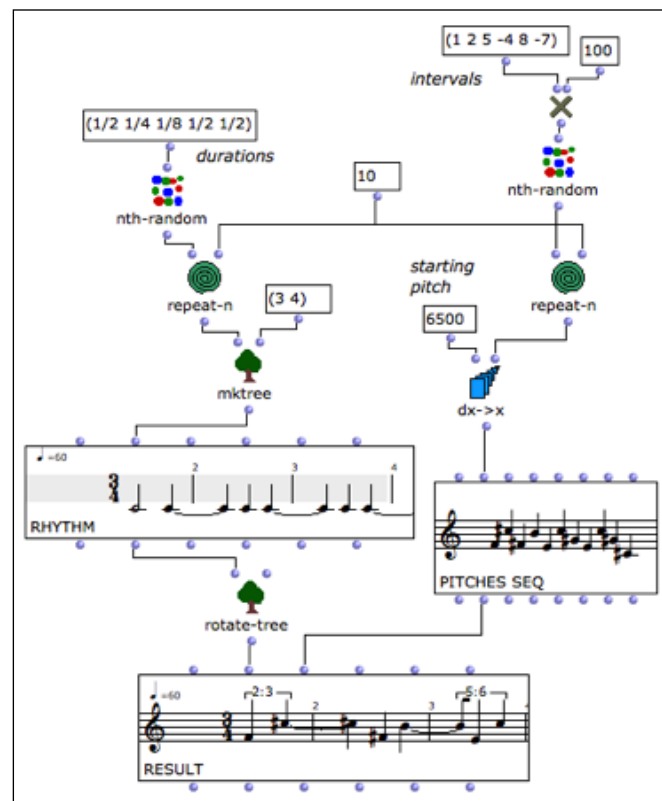


From sketching to composing

« General » music making:

Sketch (record, draw...)

Edit

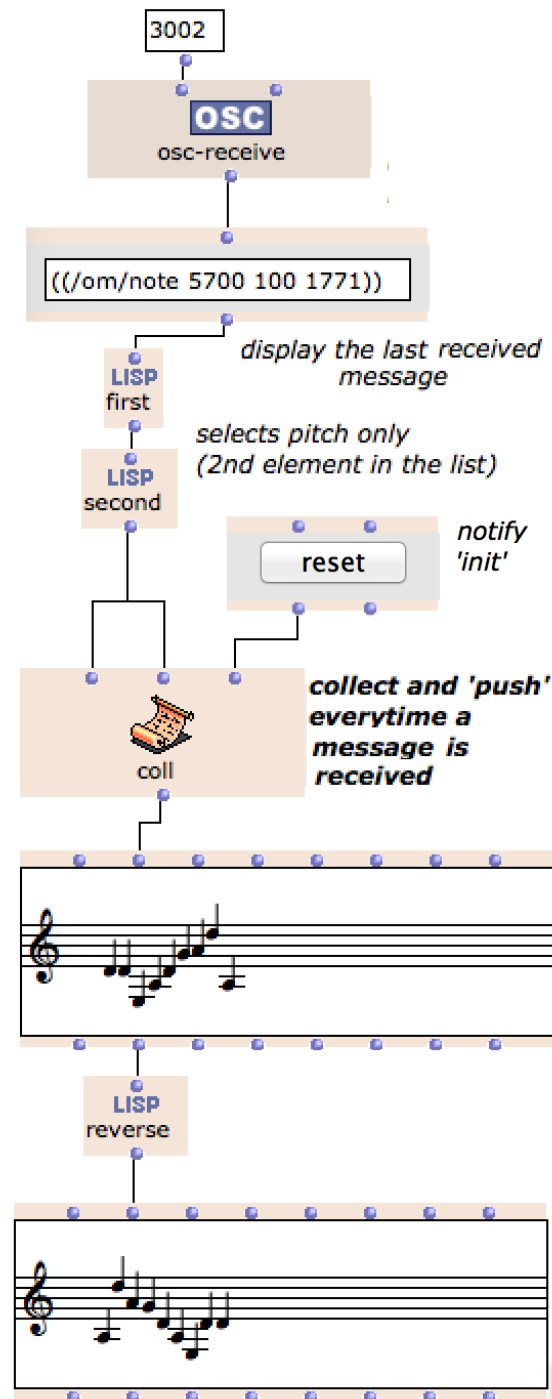
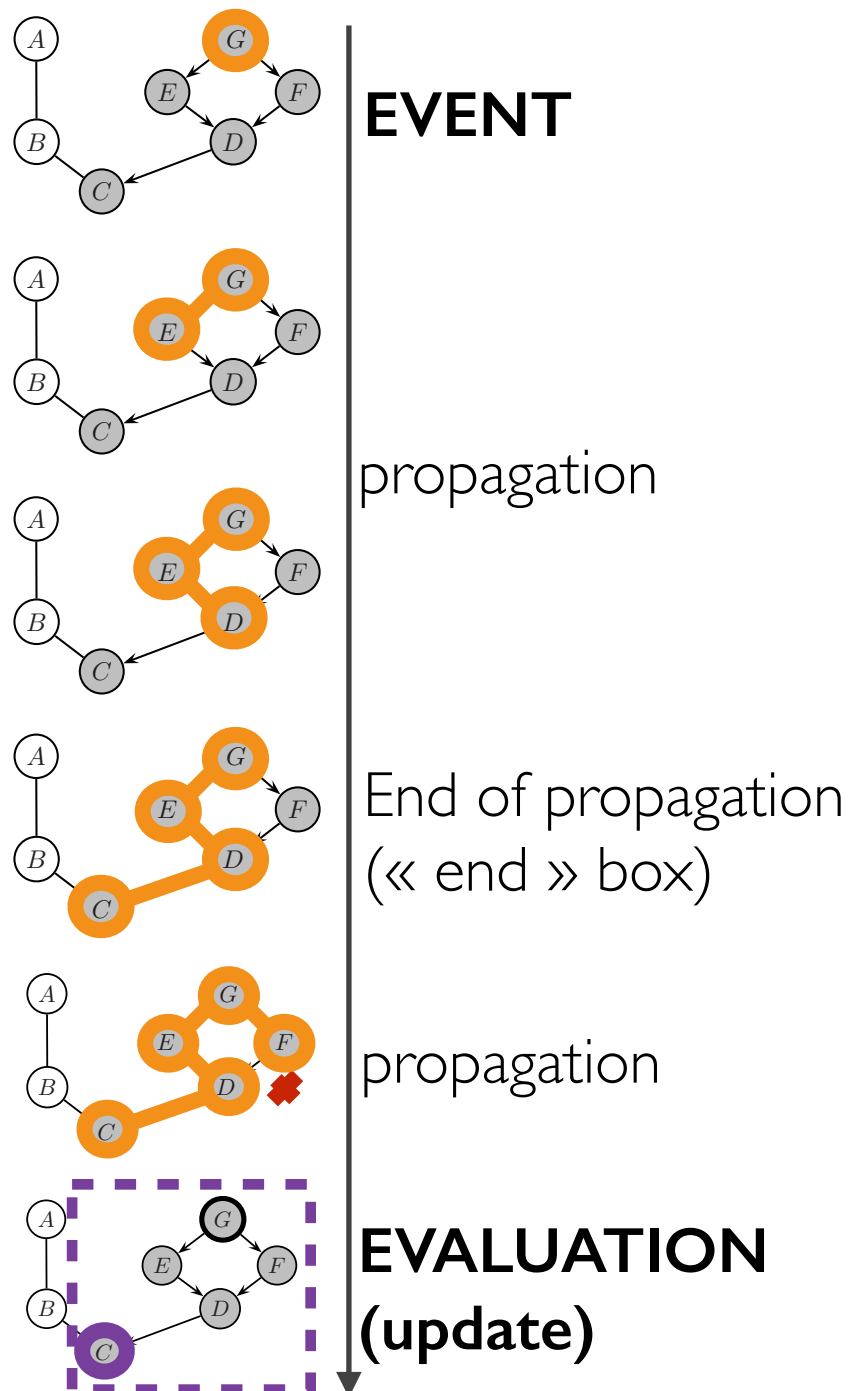


Computer-aided Composition:

Formalize

Compute

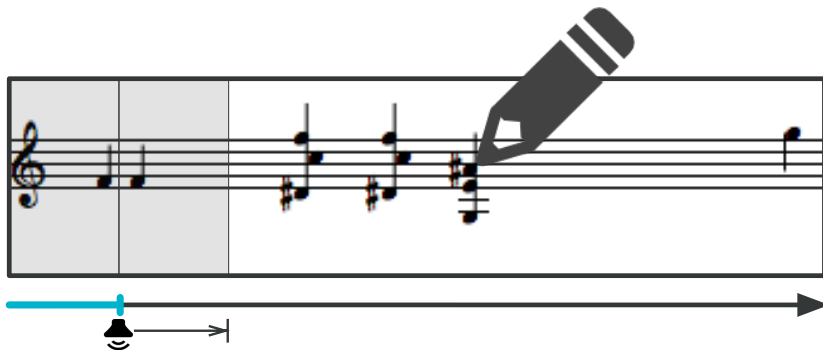
Teaching with Reactive OpenMusic



- Propagation of changes down the tree
- Immediate feedback
- Better understanding of the compositional process
- Tweak and Freeze

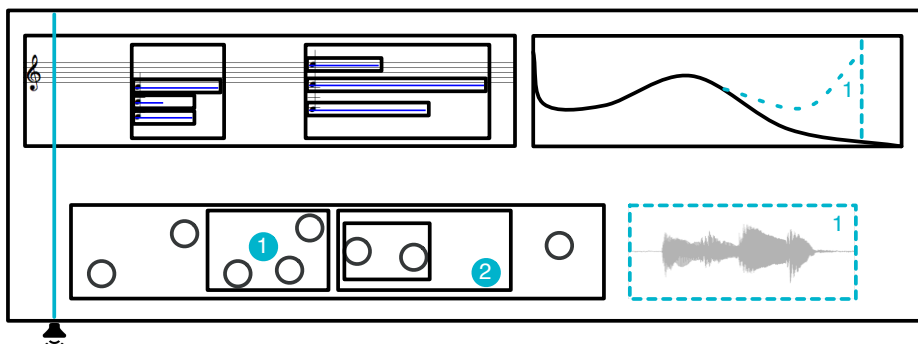
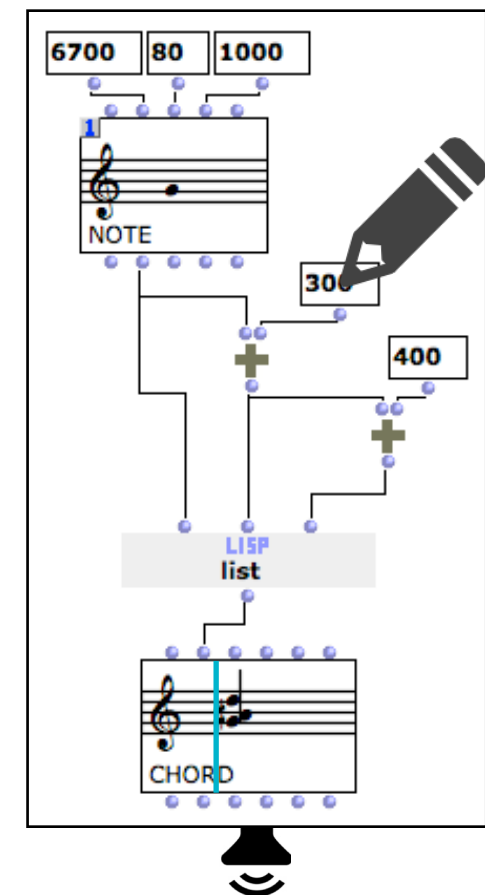
- J.Bresson, J-L.Giavitto « **A Reactive Extension of the OpenMusic Visual Programming Language** », Journal of Visual Programming Languages and Computing, 2014

Learning with new CAC tools



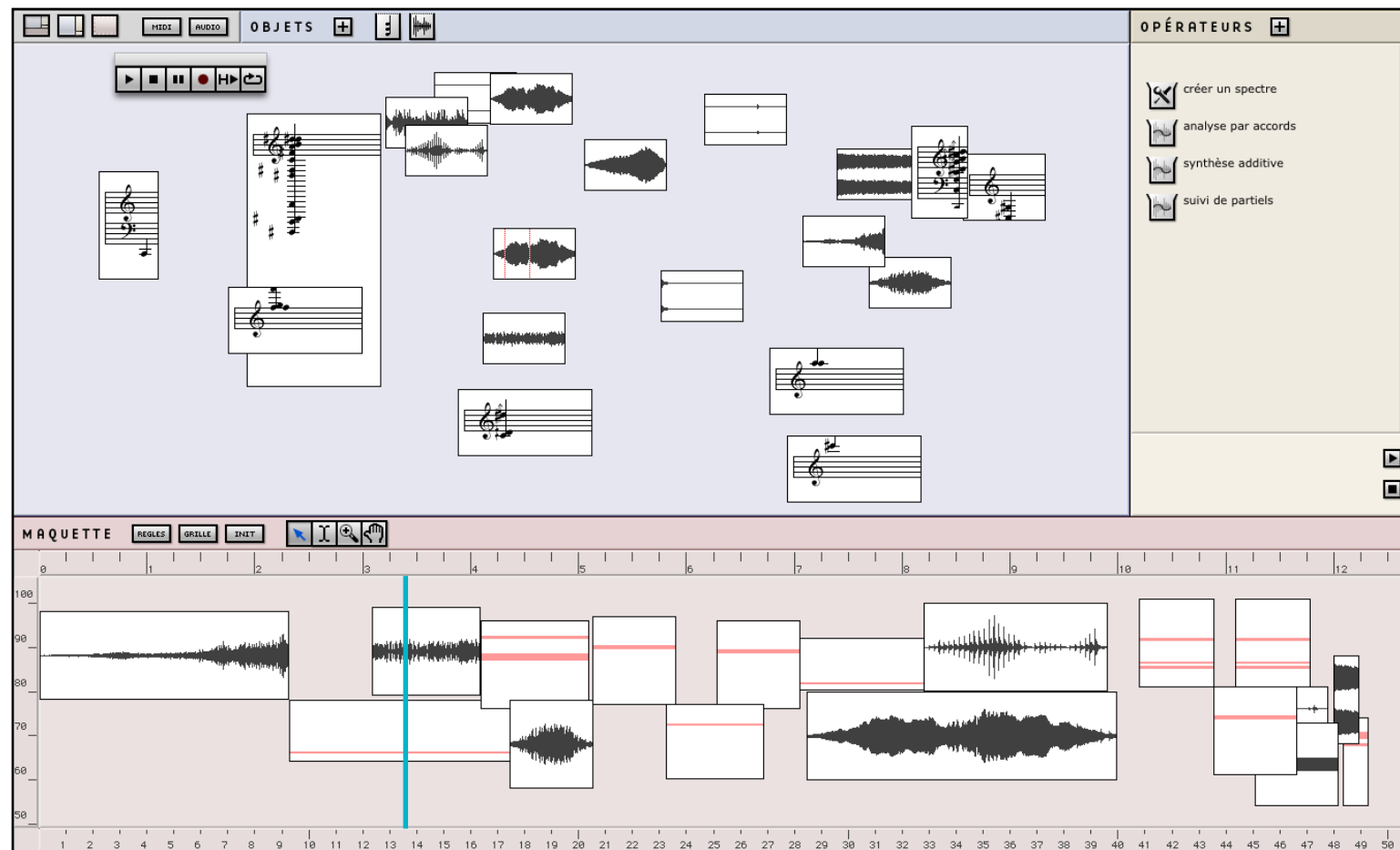
- Edition of data being rendered (sketching)

- Edition of the parent process of an object being rendered (tweaking the process)



- Integration of processes in a temporal scenario

Perspective: Musique Lab 3



- Make use of the new scheduler capabilities, for example:
 - Permanently looping sequencer,
 - Add the scheduling API in the object list,
 - Allow connection with the external environment...

Conclusion

- Partnership with the French Ministry of Education (Musique Lab 2):
 - higher level than the patching environment
 - teach music theory and how some musical pieces were created
- OpenMusic demonstrated to be useful for learning programming:
 - composers end building their own methods, objects... and libraries!
 - non-Lisp developers easily understand and get involved in the development
- New technology for extended computer-aided composition helps to:
 - better understand compositional processes
 - bridge between general audience software and technical CAC