Advanced real-time sound and data processing with FTM&Co

NOTAM Workshop 24.–28.1.2011

presented by Diemo Schwarz

IMTR Real-Time Music Interaction Team
Ircam–Centre Pompidou

http://ftm.ircam.fr

FTM 2.5.0 release beta 19
for Max 5 on Mac OS X

ircam
Centre Pompidou

Plan

- FTM
  - Introduction
  - Basics
  - Examples and exercises
  - Persistency: SDIF, SQLite
- Gabor
  - Principle
  - Examples and exercises
- MnM
  - Overview, examples and exercises
  - Gesture Follower
- CataRT

ircam Centre Pompidou

What is FTM&Co?

- a real time object system for Max/MSP
- representation of musical structures
- representation of sound data and structures
- abstraction from the real-time system (Max, PureData, ...)
- Timeline:

- Reference article:

Norbert Schnell, R. Borghesi, D. Schwarz, F. Bevilacqua, R. Müller « FTM — Complex data structures for Max », International Computer Music Conference (ICMC), Barcelona, 2005

ircam Centre Pompidou

Introduction

FTM_class_001

- Un système d'objects temps réel pour Max/MSP

Actuellement:

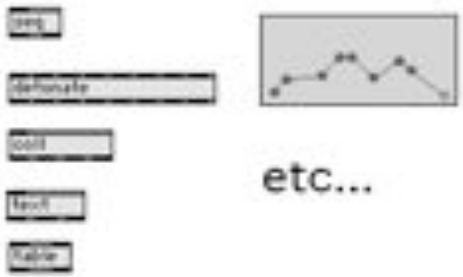
nombres 4 3.141593

symboles midi a3 '60 64 67'

listes 60 64 67

strings (chaînes de caractères) "60 64 67" "note on"

Comment représenter et «manipuler» des structures de données plus complexes???



etc...

On ne peut pas utiliser ces structures de données dans un flux de calcul!!

Comment représenter et des "listes" à plusieurs niveaux???

[[60 64 67 72] [62 65 67 71]]

cela n'était pas encore possible!!!

courtesy of Mikhail Malt

FTM Motivations

- Audio analysis/synthesis (Gabor)
  - representation of grains, waves/periods and frames
  - representation and organisation of sound descriptors
- Gesture analysis, classification and recognition (MnM)
  - (linear) algebra and matrix processing
  - representation and (statistical) modelling of gestures gestalts
- Score representation and following (FTM, suivi)
  - representation of a score or time series of data
  - representation of objects in a score

➡ modularity & flexible structuring of memory

FTM Components

- FTMlib shared library for Max
  - message system extension
  - introducing complex data types
- + Max externals
- + Editors and graphical externals

FTMlib Features

- Objects of predefined classes
  - mat, dict, track, fmat, expr, bpf, ...
- Expression evaluation
- File import/export (soundfiles, MIDI, SDIF)
- Object serialization and persistence

Terminology

Real World	Object-Oriented Programming	FTM	Max
a *thing* with a name, state, and properties *(an alert black-and-white dog named Otto)*	object	(data) object	(some specific Max objects = externals)
an *action* and *reaction* *("sit!")*	method and return value	message and return value	message (and output)
a *process* *(the vet)*	(function, process)	module (external)	Max object (external)
a *class* of similar things (*dogs*)	class	class	–
a more *general* or more *specific* class (*pug dogs, animals*)	superclass, derived class	– (types, related classes)	–

FTM Concepts

- separation of *data* and *operators*

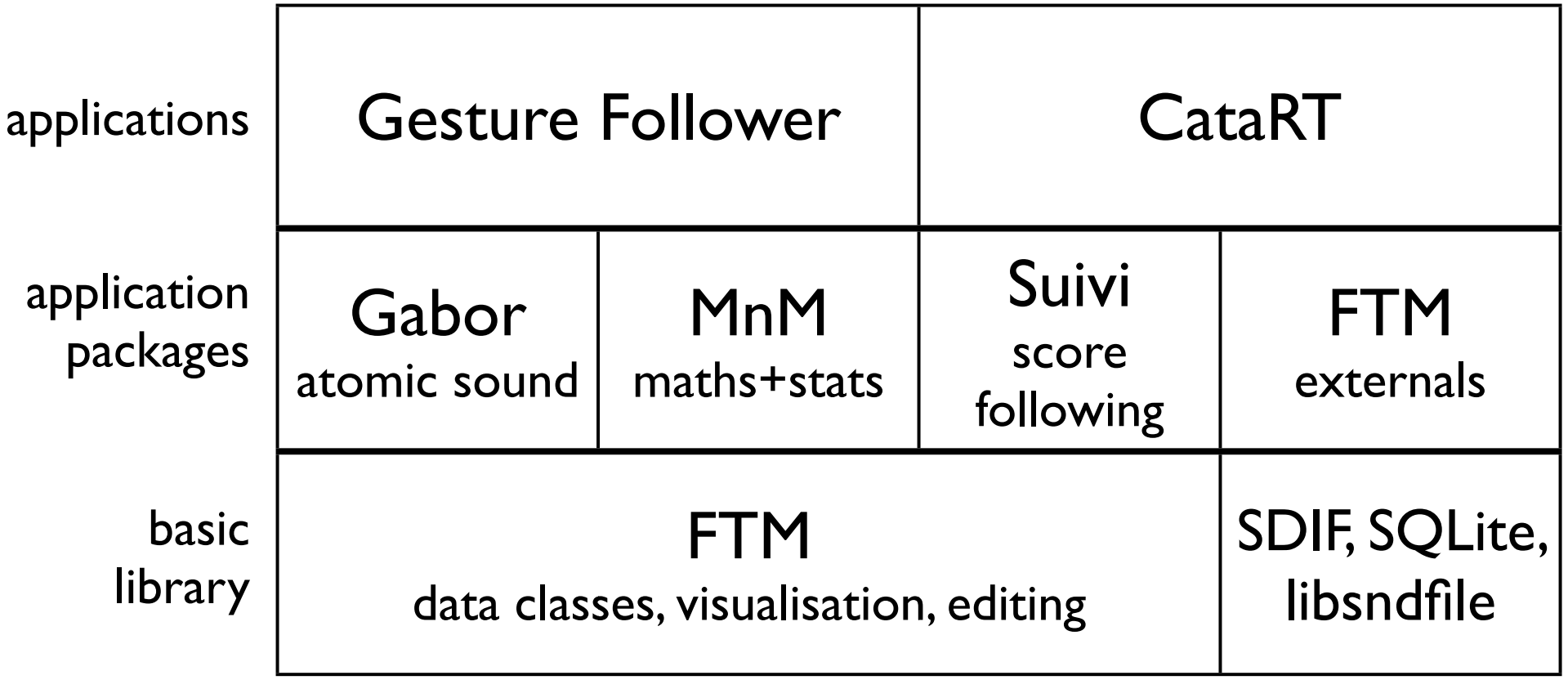
➡ *data*

- objects of optimised classes with methods
- file import & export (text, audio files, MIDI files, SDIF)
- persistence (save within patcher and in separate files)
- conversion to Max values and lists
- sent within Max values and lists (as *references*)
- simple garbage collector (by reference count)

➡ *operators*

- FTM extended message box
- FTM externals

The FTM Package Familiy



http://ftm.ircam.fr

http://imtr.ircam.fr

ircam Centre Pompidou

Distribution

- FTM 2.5 current stable versions for Max5
  - FTM 2.3 and 2.4 for Max 4.6
- FTM is free (*libre*) open source software (LGPL)
- Gabor&MnM are free (as in *free beer*)
  - both are distributed together
- Web site http://ftm.ircam.fr
- User mailing list ftm@ircam.fr on http://lists.ircam.fr
  Developer list ftm-devel@sourceforge.net

Credits and References

- FTM&Co is developed in the Real-Time Musical Interactions team at IRCAM by Norbert Schnell and Riccardo Borghesi, Diemo Schwarz, Frederic Bevilacqua, Remy Müller, Jean-Philippe Lambert
- with contributions by Julien Bloit, Baptiste Caramiaux, Arshia Cont, Pierre Duquesne, Sebastien Gulluni, Baptiste de la Gorce, Bruno Zamborlin
- Reference article:

  Norbert Schnell, R. Borghesi, D. Schwarz, F. Bevilacqua, R. Müller « FTM — Complex data structures for Max », International Computer Music Conference (ICMC), Barcelona, 2005

ircam Centre Pompidou

Gabor:

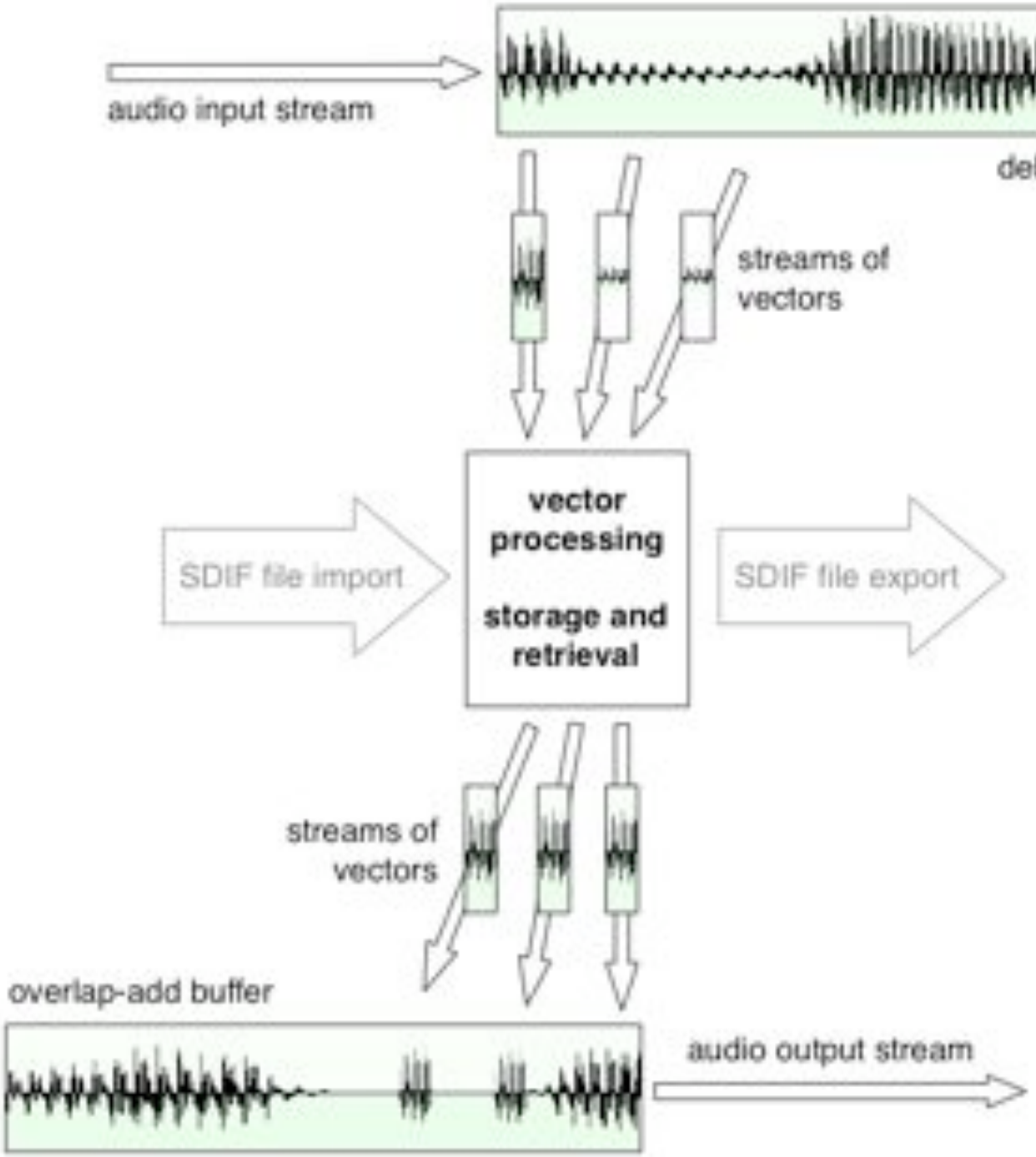
- Dennis Gabor, *Acoustical Quanta and the Theory of Hearing*, Nature, 1947
- *Multi-Representation*
  - grains, waveforms, spectral frames, partial sets
- *Real-Time*
  - extension modules for Max/MSP, PD
- *Analysis–Synthesis*
  - ...and transformation in a unified framework
- Reference article:

  Norbert Schnell et al. « Gabor, Multi-Representation Real-Time Analysis/Synthesis », COST-G6 Conference on Digital Audio Effects (DAFx), Madrid, 2005

ircam Centre Pompidou

Two Fundamental Ideas...

1. Atomic sound particles in various descriptions
   - *time-domain*: grains, wave periods
   - *spectral domain*: FFT frames, phase vocoder frames, sinusoidal partials
   - *metadata*: cepstrum spectral envelopes, filter coefficients, sound descriptors
2. Arbitrary rate processing
   - picking grains
   - pitch synchronous
   - multiple frame rates



...their Realisation...

1. Unified framework
   - sound particles represented as generic data objects
     - matrix of floating point numbers
     - streams as time-sequence of objects
     - (others: hash-table or matrix of objects)
   - sent in between specific analysis or synthesis modules
2. Sound processing via message passing (control)
   - 64 bit time tags for sub-sample precise reconstitution of the signal
   - input/output buffers with interpolation

...and their Advantages

1. Unified framework allows generic manipulation, storage, and display of atoms
   - higher modularity
   - can use math
   - no longer needing a monolithic

     analysis+synthesis+manipulation module
2. Arbitrary control rate processing
   - is adaptable to frequency/texture/rhythm of sound or analysis algorithm
   - not limited to fixed-rate audio DSP chain

From Signal to Atom

- Ways to cut the input buffer into vectors:
  - arbitrary grain extraction: `gbr.grain~`
  - pitch synchronous elementary waveforms: `gbr.psy~`
    - based on *yin* pitch estimation algorithm
      (de Cheveigné and Kawahara, JASA 2002)
  - constant rate frames: `gbr.slice~`
    - FFT
    - phase vocoder analysis
    - additive analysis

From Atom to Signal

- Windowing
- Inverse FFT
- FFT^{-1} (additive synthesis)
- Overlap add: `gbr.ola~`
  - shortcut for paste and drain (output delay line buffer)

...and in between...

- Manipulation
  - of the data
  - of the timing
- Display
- Storage and retrieval
  - files (audio, SDIF)
  - databases
- Future: network or inter-application send/receive via OSC

Gabor Summary

Ideas	**Realisation**
atomic sound particles in multiple representations	unified framework of specific analysis–synthesis, generic data
arbitrary rate processing	event processing with time-tags + delay lines

MnM package

"Mapping is not Music"

- Classification and recognition

  motion capture data, sound and music

- Basic matrix calculations
- Linear algebra
- PCA, GMM, HMM, NMD, EM, etc.

- Reference article:

  Frederic Bevilacqua, R. Muller, N. Schnell « MnM: a Max/MSP mapping toolbox », New Interfaces for Musical Expression, Vancouver, 2005

Richard Siegal - *If/Then* Dance Performance

Mubu

- Packaging of frequently used signal analysis, representation and synthesis algorithms with perfect synchronization as MSP externals
- Mailing list mubu-beta@ircam.fr
- Reference article:

  Norbert Schnell, et al. R. Borghesi, D. Schwarz, F. Bevilacqua, R. Müller « Mubu & Friends - Assembling Tools for Content Based Real-Time Interactive Audio Processing in Max/MSP », International Computer Music Conference (ICMC), Montreal, 2009

Real-Time Corpus-Based Concatenative Synthesis:

CataRT

http://imtr.ircam.fr

Principle of CBCS

- **Corpus-based**
  sounds + segmentation + **descriptors**
  (timbre, metadata, classes)
- **concatenative**
  grains + overlap + chaining
- **real-time synthesis**
  interactive + explorative (navigation)

Motivation and Approach

- work with all the nuances of *real* sound
- large sound databases exist, ready to use
- new method → new sound creation
- *data-driven* vs. *rule-based* approach
- general superiority of data-driven approaches in many other domains

Influences and Links

- Inspired by concatenative speech synthesis
- CBCS in real-time (as implemented in CataRT) can be seen as a content-aware extension to granular synthesis:
  - providing direct access to specific sound characteristic
  - not restricted to selection by position in a sound

Applications of CBCS

- Interactive exploration of sound databases (*browsing*)
- Resynthesis of an audio target (*mosaicing*)
- Texture synthesis (*ambiences*, *soundtracks*)
- High-level instrument synthesis (example: *Synful*)
- Expressive speech synthesis

A Short History of Data-Driven Synthesis

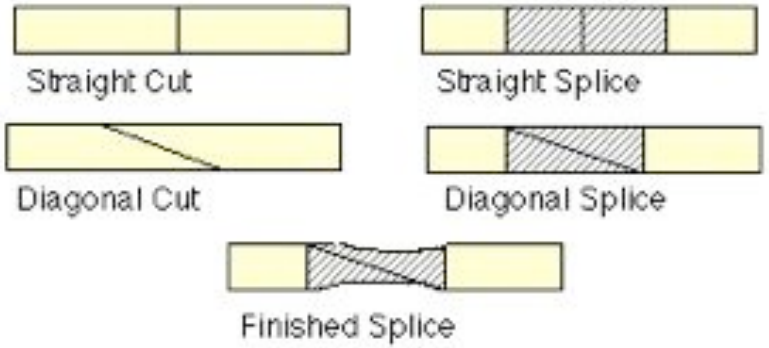
I. Prehistory (1948)
- Concrete and early electronic music

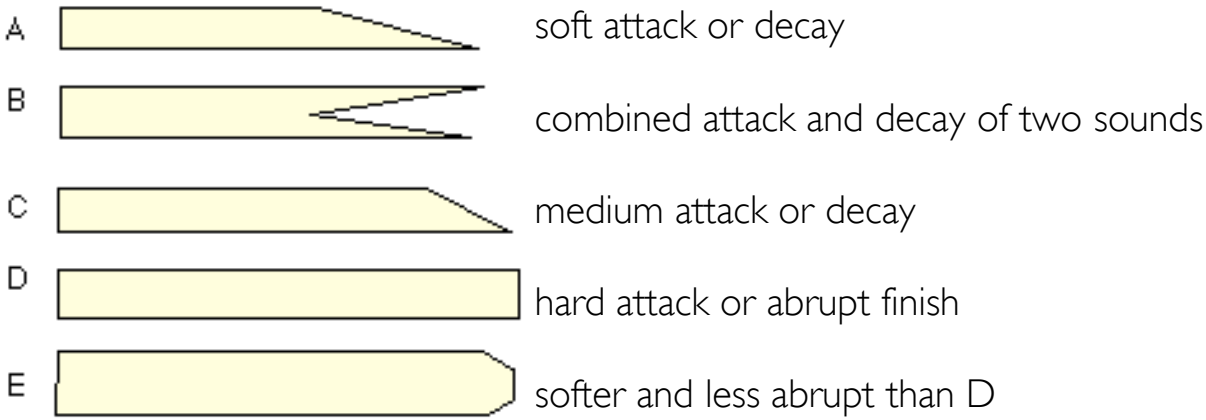
II. The Classics — revisited (1980, 1990)
- Sampling
- Granular Synthesis
- John Oswald

III. The Modern Times (2000)
- many research projects, often in mutual ignorance

Tape splicing

Karlheinz Stockhausen:
Étude des1000 collants (1952)

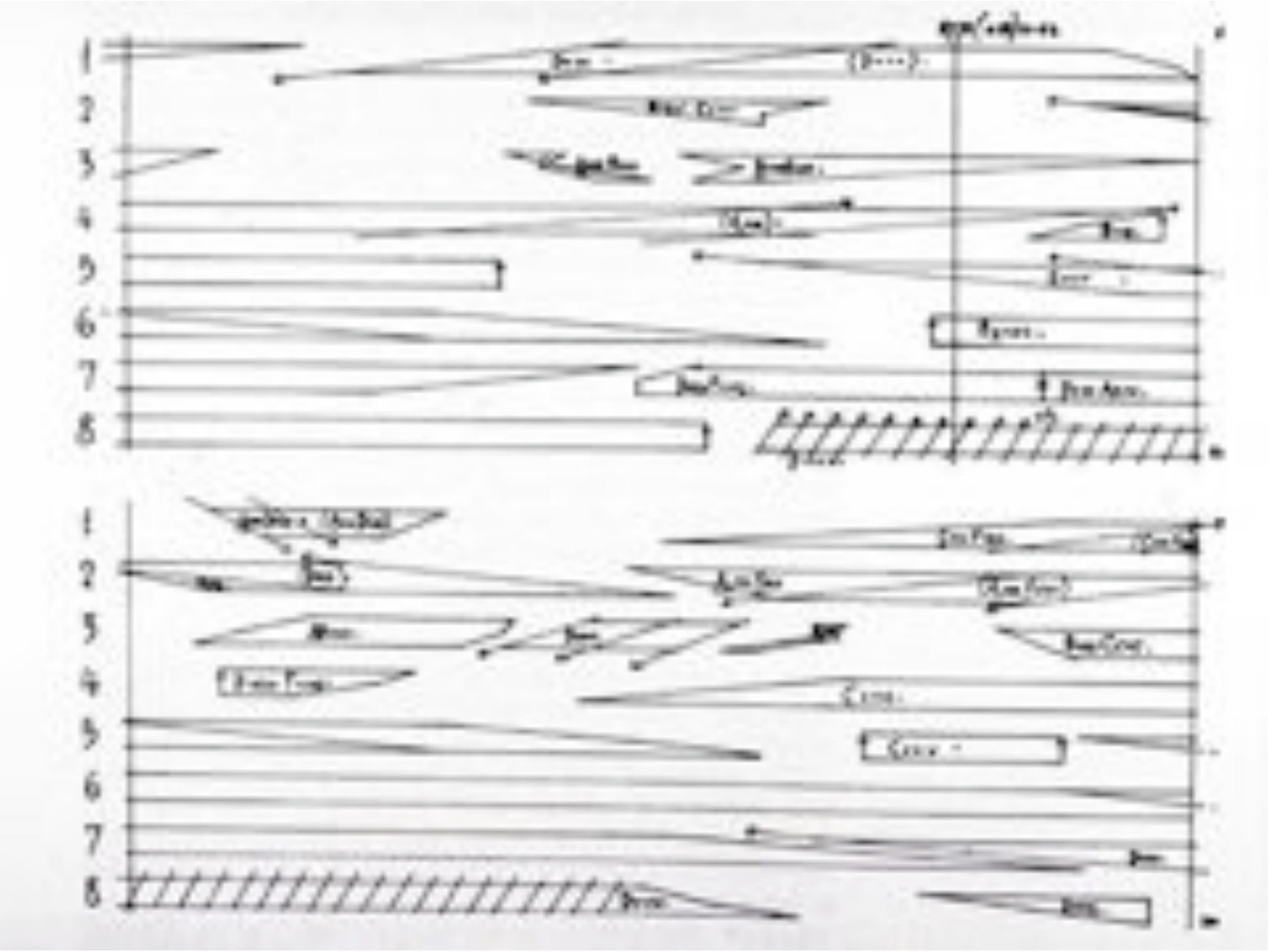




[figures: Mikhail Malt]

ircam Centre Pompidou

John Cage: *Williams Mix* (1953)

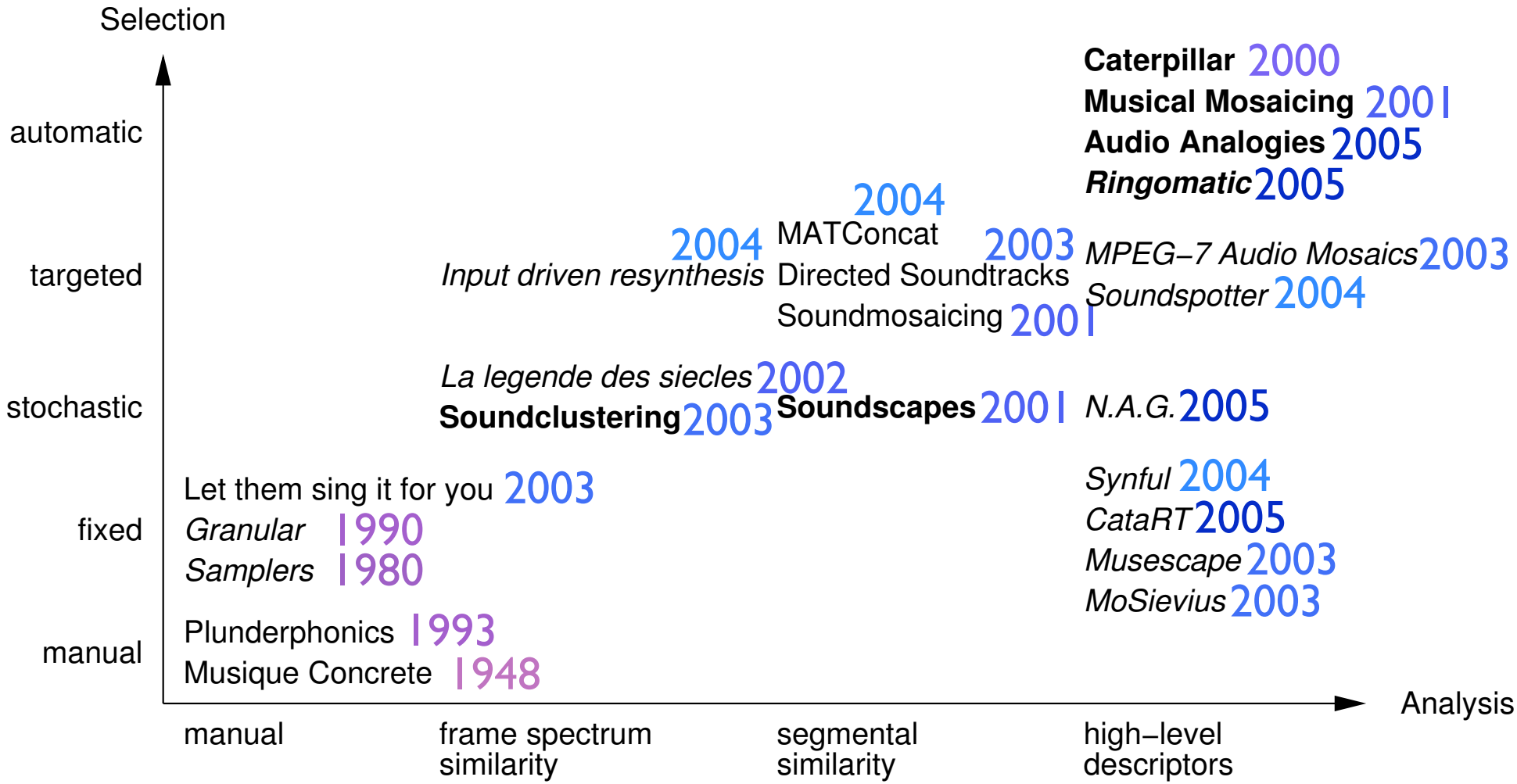


[http://www.medienkunstnetz.de/works/williams-mix]

ircam Centre Pompidou

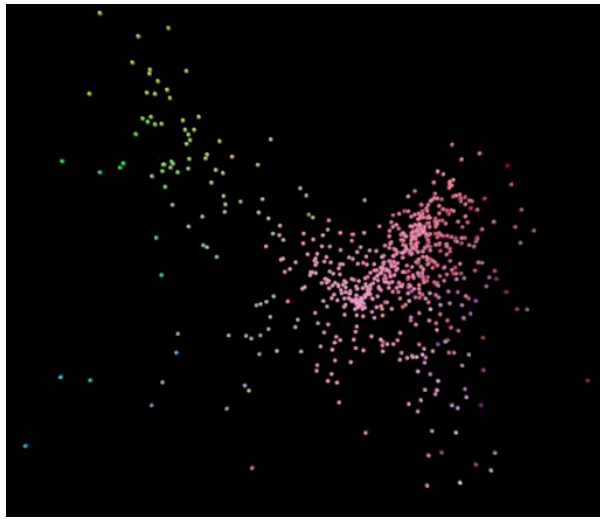
Related Work up to 2005

[Schwarz, JNMR 2006]



ircam Centre Pompidou

Current Related Work



- *TimbreID* (William Brent) is a collection of PD externals that allow descriptor analysis, corpus-based concatenative synthesis and CataRT-style 2D interaction
- *Extending voice-driven synthesis to audio mosaicing* (Jordi Janer, Maarten de Boer): a system for controlling audio mosaicing with a voice signal
- *Mused*: Navigating the Personal Sample Library (Graham Coleman): a system for sample-based music composition
- ScrambledHacks (2006): performance system for spectral lookup alongside ganged music video concatenation

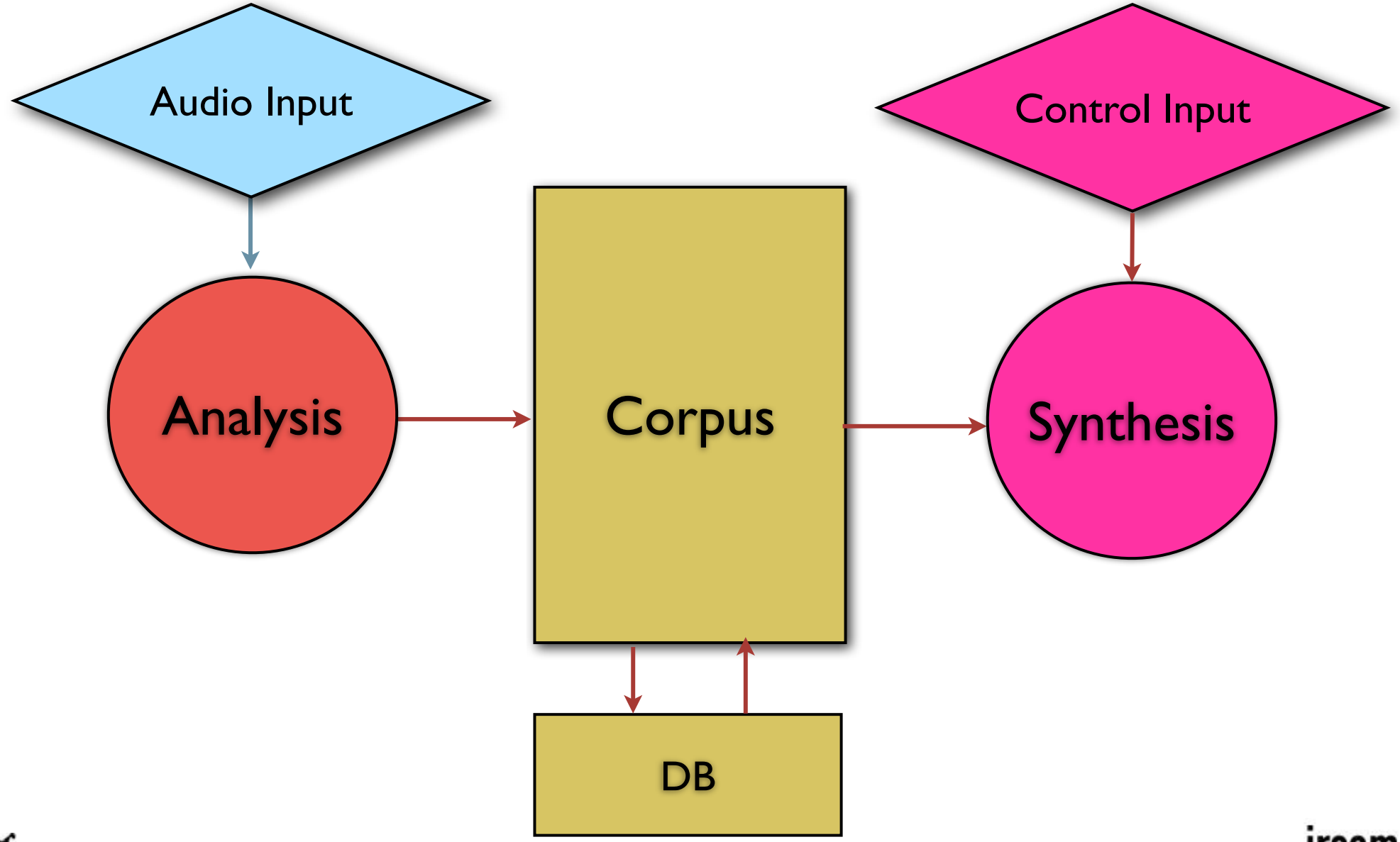
CataRT Introduction

- flexible testbed to experiment real-time corpus-based concatenative synthesis: **CataRT**
- realisation of CBCS in real-time
  - patch for Max/MSP with FTM, Gabor, MnM by Norbert Schnell et al. from **http://ftm.ircam.fr**
  - released as Free Software under GPL at **http://imtr.ircam.fr**
  - mailing list **concat** on **http://list.ircam.fr**
  - modular MVC architecture: building blocks for you to use

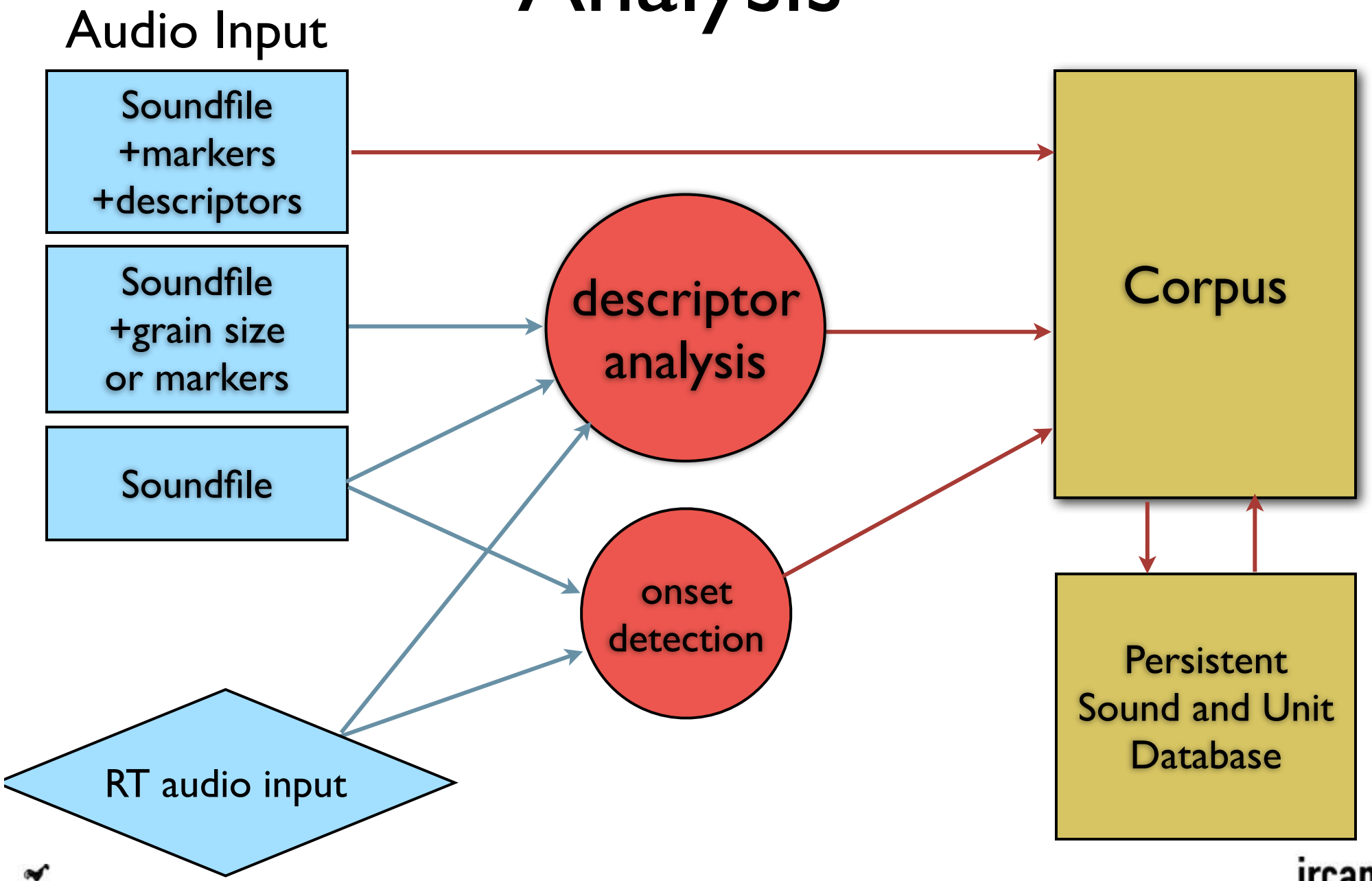
~~CataArt (?)~~

CataRT

CataRT Overview

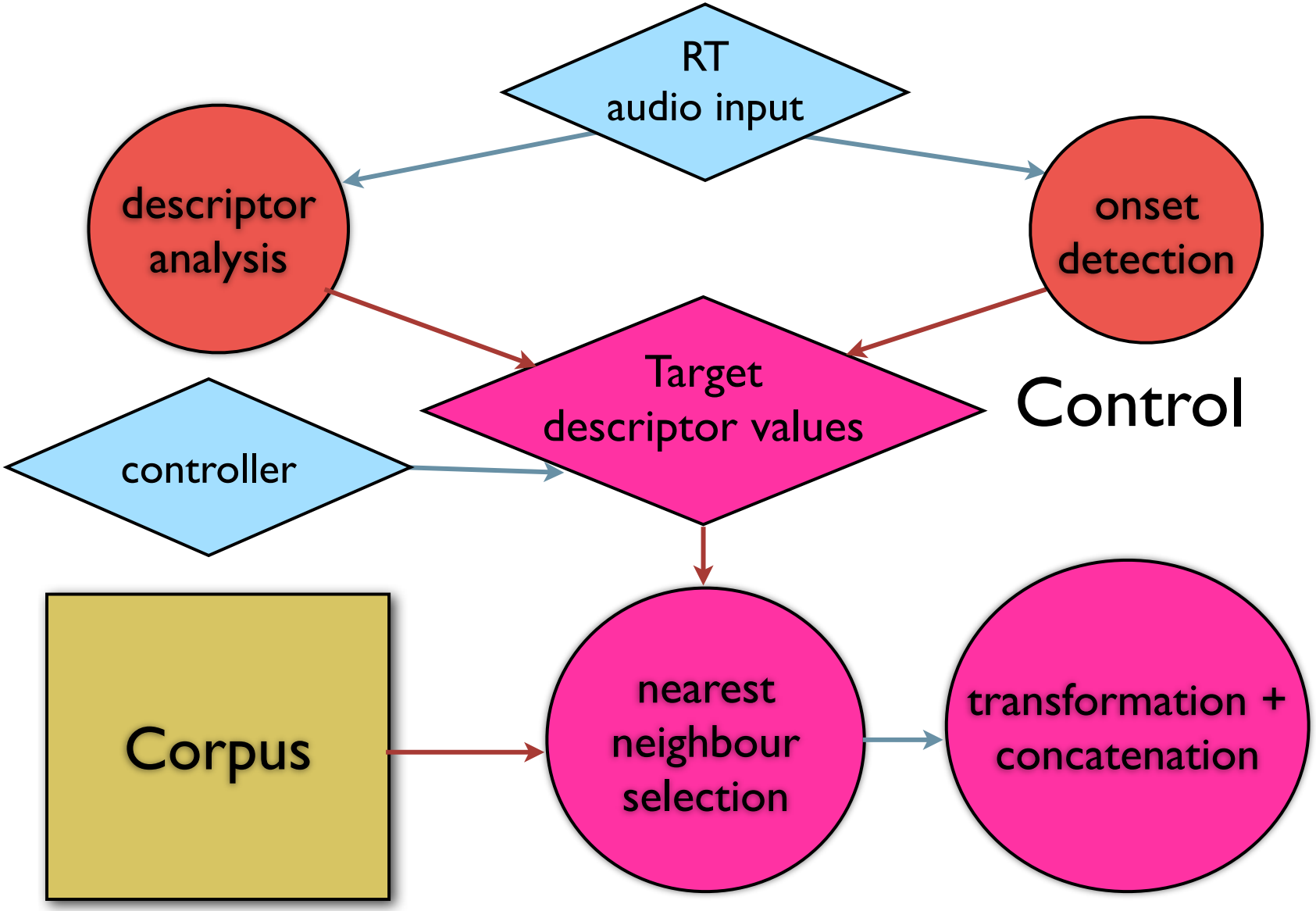


Analysis



ircam Centre Pompidou

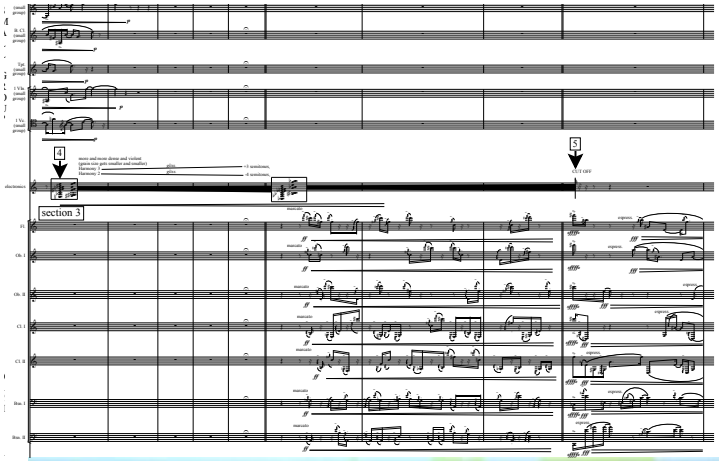
Synthesis



ircam Centre Pompidou

Composers and Projects using CataRT

- Sound design and composition
  Luis Naõn, Hans Tutschku, Matthew Burtner, Sebastien Roux, Hector Parra, Roque Rivas, Ben Hackbarth
- Installation
  Plumage (project *Enigmes* about the navigable score, Roland Cahen), Pierre Jodlowski, Cécile Babiole, Franck Leibovici
- Live electro-acoustic music
  Dai Fujikura, Stefano Gervasoni, Miguel Angel Ortiz-Pérez, Sam Britton, Aaron Einbond, Bruno Ruviaro, Christopher Trapani, Marc Vitoria, Eric Maestri, *theconcatenator*
- Examples: **myspace.com/catartsoftware**



Installation *GrainStick*

- "Rainstick" gestural control metaphor
- Demo of *SAME (Sound And Music for Everyone Everyday Everywhere Every way)* EU project
  - Composition of sounds and ambiences by Pierre Jodlowski
  - WFS spatialisation and Motion Tracking by Grace Leslie and the Room Acoustics team
  - Gesture analysis by Bruno Zamborlin and Diemo Schwarz

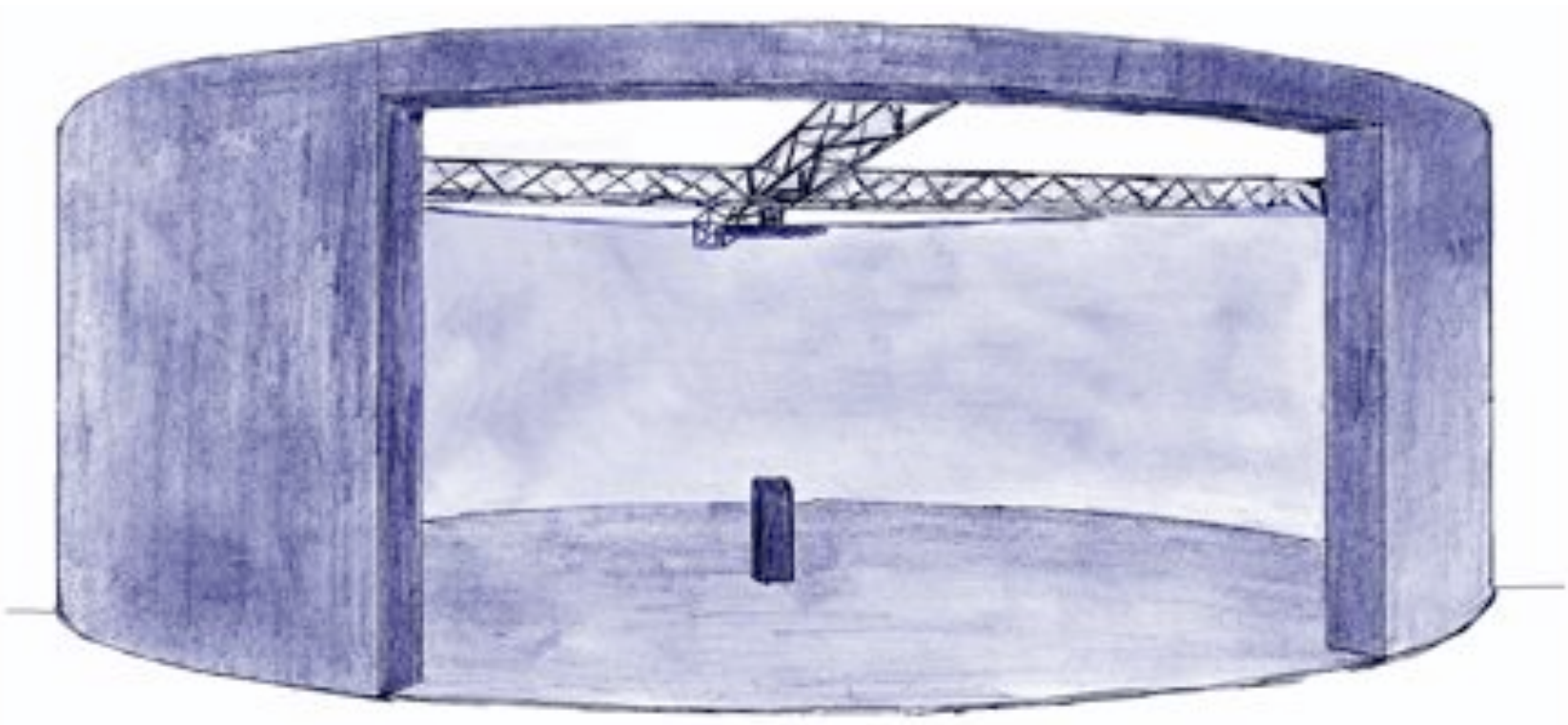
Grainstick Example 1

Grainstick Example 2

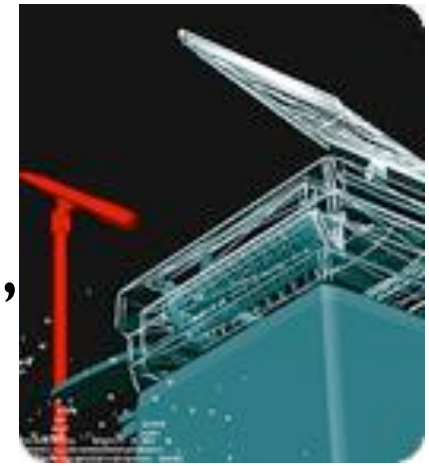
ircam
Centre
Pompidou

Grainstick Installations

- Prototype: Agora Festival 2009, Ircam, Paris
- Public installation: 6. June 2010, *Cité des Sciences*, Paris
  - including video projection, more collaborative elements, several universes



Installation *Xe-Rocks*



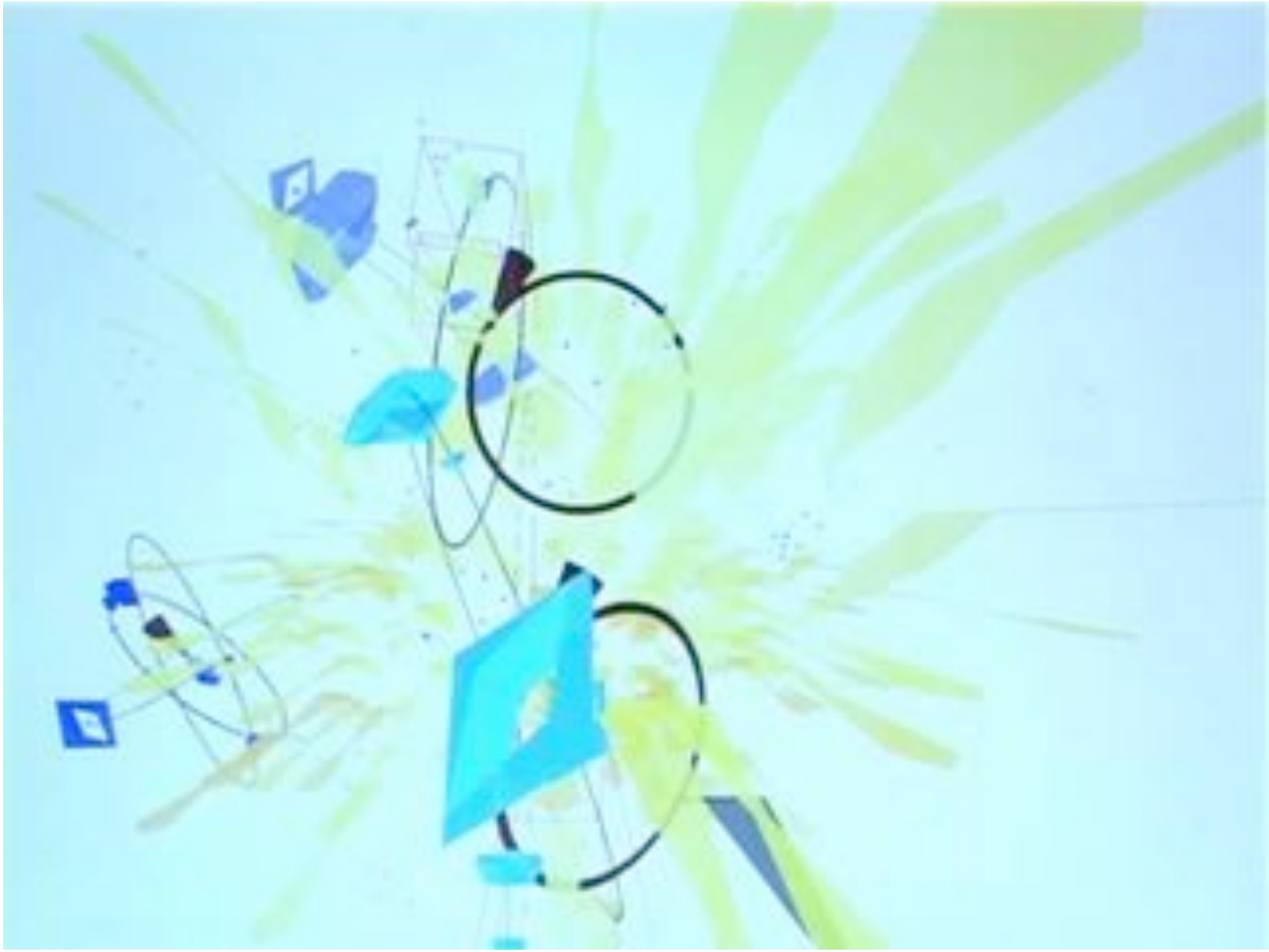
- Turning photocopiers into musical instruments, by video and sound artist Cécile Babiole
- On show at the ECM Gantner, Bourogne

Installation *Plumage*

- Collaboration with national design school ENSCI led by Roland Cahen, Yoan Ollivier, Benjamin Wulf (ENSCI), Christian Jacquemin, Rami Ajaj (LIMSI), Diemo Schwarz (Ircam)
  - within the project *ENIGMES* about the navigable score
  - 3D interface using *Virtual Choreographer* **http://virchor.sourceforge.net**

Plumage

Plumage



Plumage

Live Recording the Corpus

- Corpus-based improvisation with live sound input
  - Diemo Schwarz with Etienne Brunet
- Live Algorithms for Music (LAM) concerts:
  - Improvisation of George Lewis, Evan Parker, Sam Britton, Diemo Schwarz and others
  - *Rien du tout* by Sam Britton, Diemo Schwarz

Electro–Acoustic Music (1)

- *Whisper Not* by Stefano Gervasoni for viola and electronics, computer music realization by Thomas Goepfer
  - "response" of CataRT to the instrument
  - morphing between corpora of pizzicato viola and water-drops

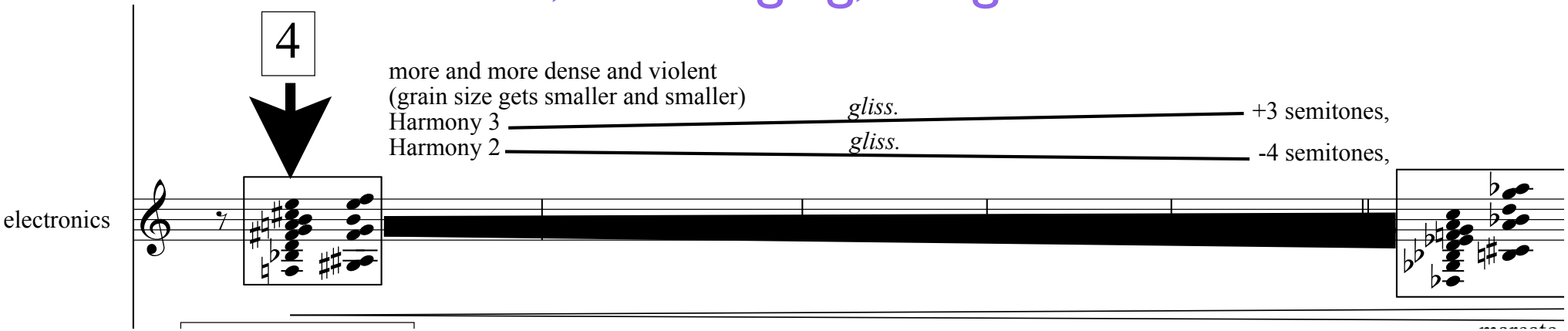
Interaction with pre-recorded sound,

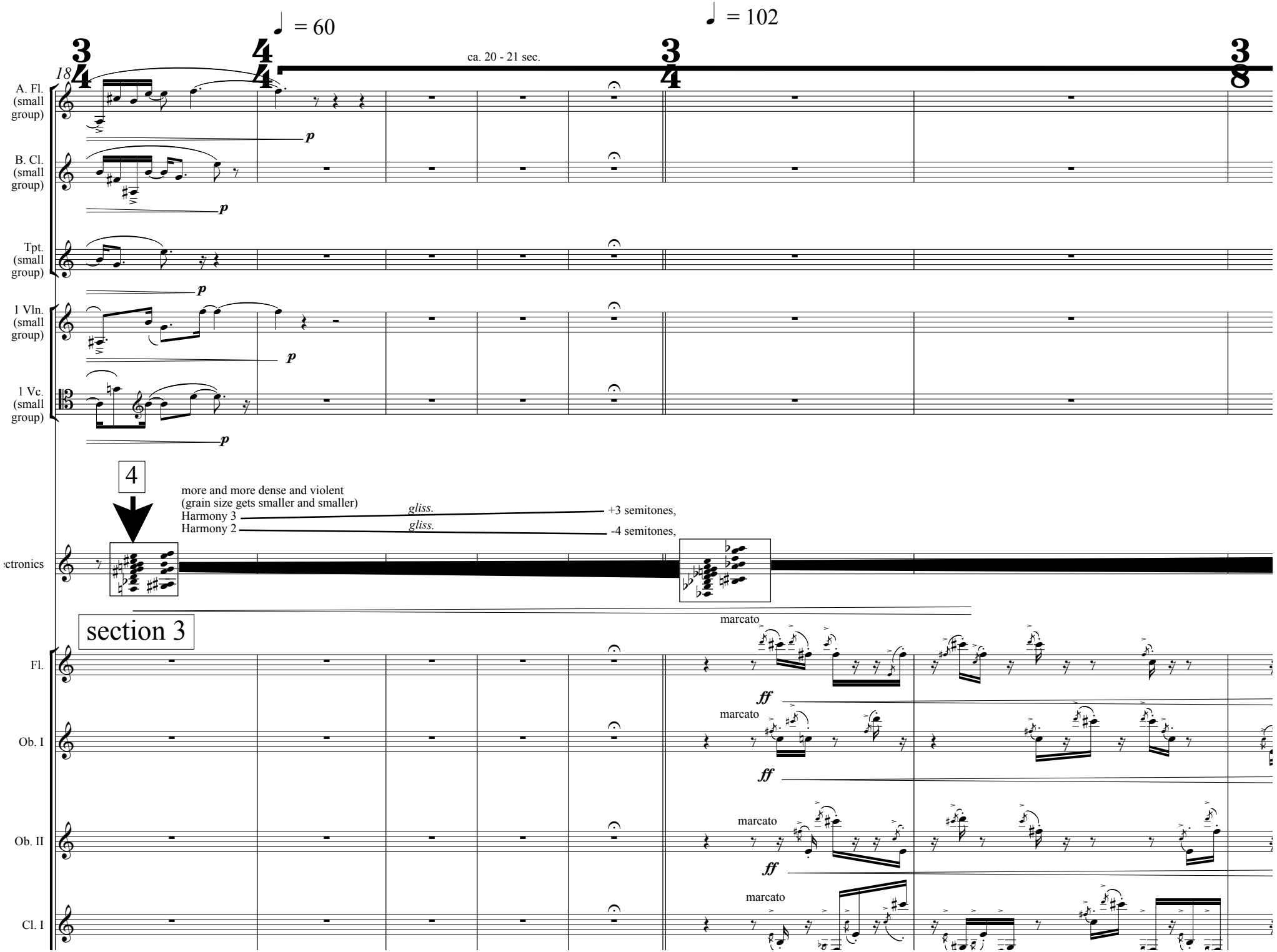
interpolation, rearranging, navigation

Electro–Acoustic Music (2)

- *swarming essence* by Dai Fujikura for orchestra and electronics, computer music realization by Manuel Poletti
  - harmonic selection of specially composed small instrument group and extended playing style
  - corpus-based orchestration, separate treatment

Re-orchestration, rearranging, navigation



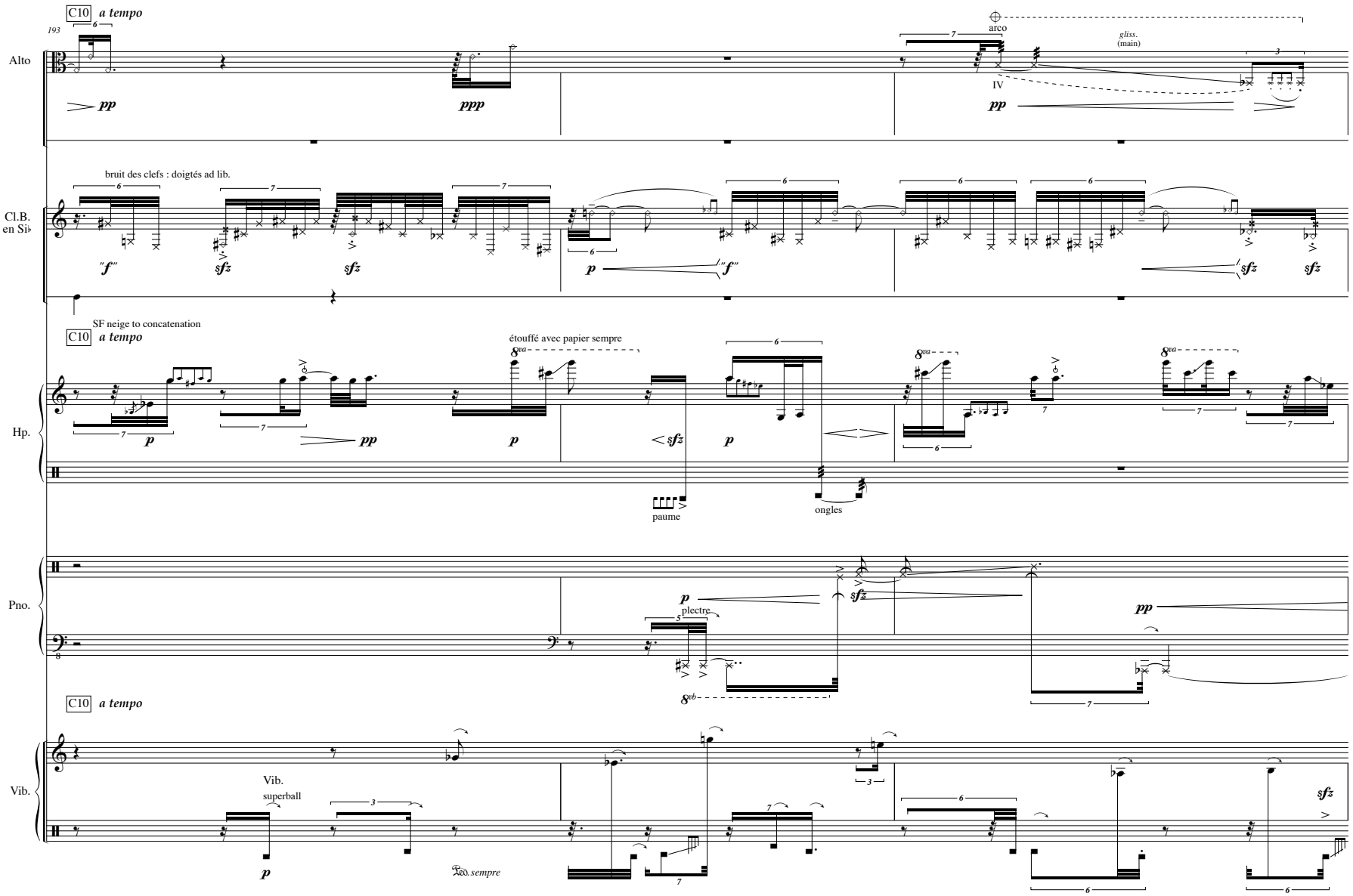

♩ = 60
♩ = 102
ca. 20 - 21 sec.
18
A. Fl. (small group)
B. Cl. (small group)
Tpt. (small group)
1 Vln. (small group)
1 Vc. (small group)
p
4
more and more dense and violent
(grain size gets smaller and smaller)
Harmony 3
gliss.
+3 semitones,
Harmony 2
gliss.
-4 semitones,
ectronics
section 3
Fl.
Ob. I
Ob. II
Cl. I
marcato
ff

Live Control and Composition

- *Beside Oneself*, Aaron Einbond, 2008 for viola and electronics:
  - real time control of CataRT from audio analysis of the viola
- *What the blind see* 2009 for small ensemble and electronics
  - real time control of CataRT from audio analysis of the viola
  - corpus-based "orchestration" or transcription
  - texture example:
    - 1. Rain on leaves
    - 2. CataRT resynthesis with extended technique instrument sounds

Aaron Einbond – *What the blind see*

1. snow melting on a metal roof
2. CataRT resynthesis with instrument sounds
3. manually edited transcription
4. live reading by ensemble



Bruno Ruviaro – *Intellectual Improperty 0.6*

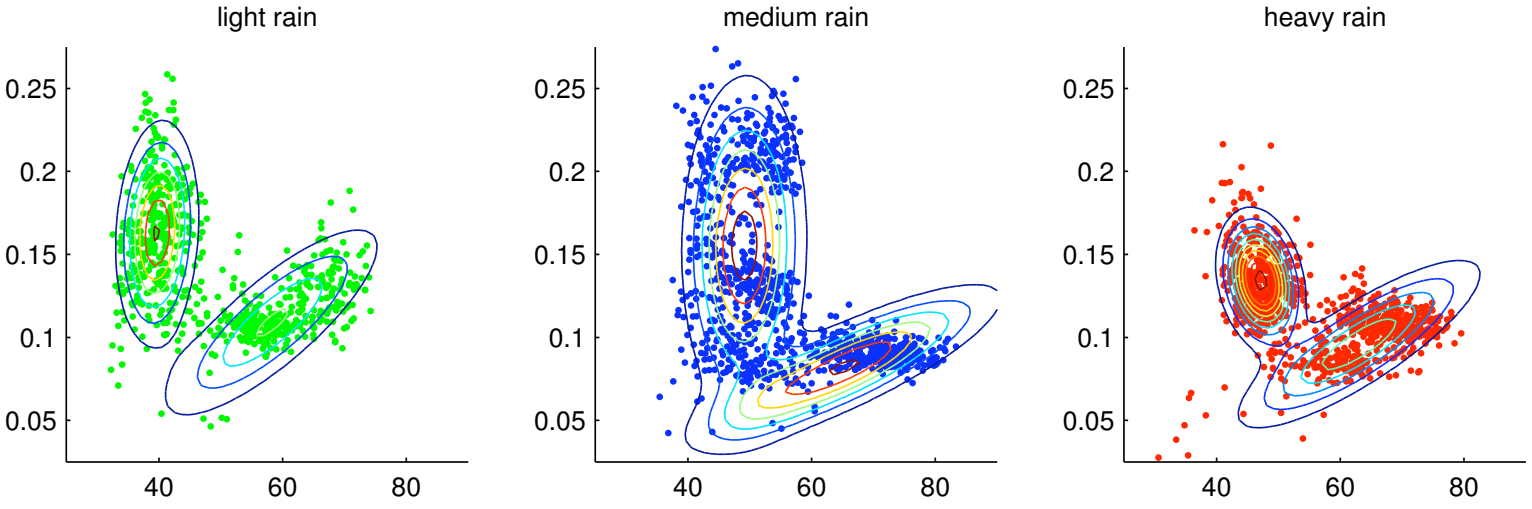
- Using *Catork*, a CataRT-based system for electronic music composition for laptop orchestras **http://www.brunoruviaro.com/catork**
  - Premiered by the Stanford Laptop Orchestra (SLOrk) April, 2010 (chamber version with six laptops), and June (full ensemble version)

New Concepts for Musical Applications

- Re-arranging units by other rules than temporal order
- Composition = Navigation through the sound space of heterogeneous sound databases
  - exploit richness of detail of recorded sound
  - efficient control by perceptually and musically meaningful descriptors
- Interaction with self-recorded sound: sound of a musician is available for interaction beyond simple repetition
- Cross-selection and interpolation: apply sound characteristics from one corpus to another, morphing between corpora
- Orchestration and re-orchestration: matching a mass of sounds to a harmonic or sonic target while retaining precise control over the result

Current and Future Directions

- Interface
  - multi-dimensional scaling by mass–spring–damper model
  - uniformisation of density
- Sound texture synthesis
- Modeling of transitions and articulations



Acknowledgements

- CataRT is based on FTM&Co by Norbert Schnell et al.
- Thanks go to:
  - Alexis Baskind, Julien Bloit, Greg Beller, Miguel Angel Ortiz Pérez
  - Roland Cahen, Yoan Olliver, Benjamin Wulf (ENSCI),
  - Christian Jacquemin, Rami Ajaj (LIMSI)
  - Norbert Schnell, Riccardo Borghesi, Fréderic Bevilacqua, Rémy Muller, Jean-Philippe Lambert (IMTR Team)
  - the French National Agency of Research ANR and the projects *Sample Orchestrator* and *Topophonie*

- Links:
  - CataRT wiki on **imtr.ircam.fr**,
  - mailinglist **concat@ircam.fr** and **ftm@ircam.fr** on http://lists.ircam.fr

ircam Centre Pompidou

Exercise Suggestions

Adapting CataRT

- Integration in your concert patch
- Adapting to your musical needs/ideas

Improving CataRT

- Corpus Data
  - soundfile editor (list/delete sounds in corpus)
  - soundset editor (define sound sets, add/remove units)
  - descriptor range viewer/editor (e.g.: exclude units below threshold)
- Analysis
  - segmentation editor
- Selection
  - constraints: don't repeat unit
- Synthesis
  - forced pitch/loudness
  - multi-channel output
- Control
  - by MIDI notes
  - by envelopes, sequencer
  - by live sound analysis

ircam Centre Pompidou

Architecture

- one `catart.data` per corpus
- one `catart.import` for segmentation and analysis
- any number of `catart.data.proxy` to access data
- any number of `catart.lcd` for display and control
- any number of `catart.selection` per corpus
- any number of `catart.synthesis` per selection or per corpus

Acknowledgements

- CataRT is based on FTM&Co by Norbert Schnell et al.
- Thanks go to:
  - Alexis Baskind, Julien Bloit, Greg Beller, Miguel Angel Ortiz Pérez
  - Roland Cahen, Yoan Olliver, Benjamin Wulf (ENSCI),
  - Christian Jacquemin, Rami Ajaj (LIMSI)
  - Norbert Schnell, Riccardo Borghesi, Fréderic Bevilacqua, Rémy Muller, Jean-Philippe Lambert (IMTR Team)
  - the French National Agency of Research ANR and the RIAM project *Sample Orchestrator*

- Links:
  - CataRT wiki on **imtr.ircam.fr**,
  - mailinglist **concat@ircam.fr** on http://lists.ircam.fr and **ftm@ircam.fr**

ircam Centre Pompidou