DISTRIBUTED PERFORMANCE OF LES SON VISIONNAIRES BY HELMUT W. ERDMANN

Georg Hajdu

ligeti center
Hamburg University of Music and Drama
georg.hajdu@hfmt-hamburg.de

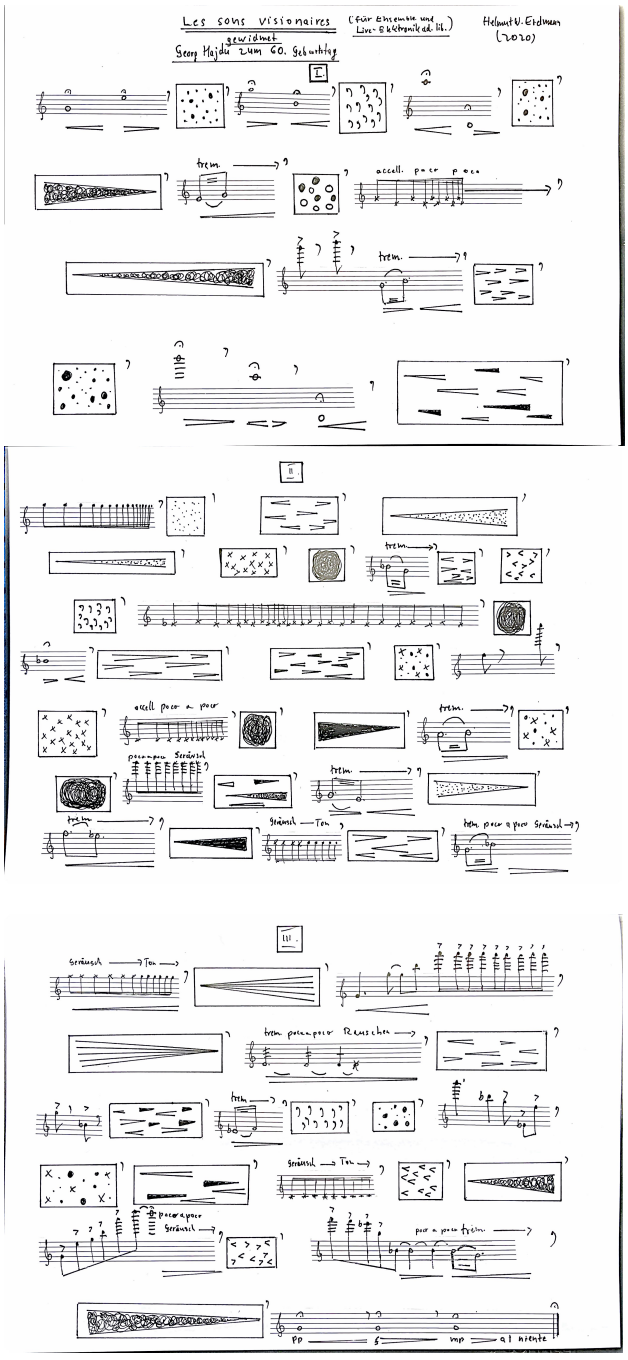
ABSTRACT

We are proposing a distributed performance of *Les Son Visionnaires* by Helmut W. Erdmann. This piece has been realized by Hajdu and his 2025 course on computer-assisted composition at the Hamburg University of Music and Drama. It features an animated score of concurrent cursors which can be controlled by either a conductor, performers or by members of the audience using a hand-held device with touch surface.

1. INTRODUCTION

Helmut W. Erdmann (*1947) is a German composer, flutist and pedagogue. His score, written in 2020 during the Pandemic, consists of 3 pages of handwritten instructions for variable ensemble featuring individual measures in standard music notation combined with boxed graphical notation. His score therefore contains elements that are supposed to be executed (standard notation) or interpreted (boxed graphical notation).

In the computer-assisted composition seminar, the score was turned into an animated score for distributed performance by employing Max, MaxScore and Drawsocket. Instead of using the scrolling score paradigm, we opted for individual cursors running across selected measures or boxes. The actions were supposed to be triggered during the performance so that a non-linear path through the score would be possible. The measures and boxes are color-coded to indicate any of the four instruments (or combinations thereof) that are supposed to perform them.



2. MAXSCORE AND DRAWSOCKET

2.1. MaxScore

MaxScore is a real-time notation package for Max developed by Didkovsky and Hajdu based on the Java Music Specification Language (JMSL) [1][2]. MaxScore has been extended to accommodate a large range of notation styles from standard Western notation to proportional and graphical music notation and used in a variety of performance scenarios [3]. MaxScore is firmly embedded in the Max environment by exchanging messages in real-time for the generation of graphics and sound. It also features *Picster* which consists of a set of drawing instruction for the generation of graphics. Picster thus allows the creation of Picster elements—individual graphical elements that can be referred to by their ID.

2.2. Drawsocket

Drawsocket is a Node.js-base Max package that implements a web server to exchange data between Max and web browsers [4]. It allows the construction and modification of web pages via Max dictionaries. Clients connect to the web server via a web-socket connection, the default port being 3002. Drawsocket employs a JSON-based format for SVG elements, HTML forms, style sheets as well as JavaScript functions and returns user actions to the Max environment via OSC bundles. Drawsocket also implements various node.js packages for display, animation and the generation of sound.

2.3. Expressions

To ensure maximum compatibility with Drawsocket, a Picster Elements adapts the Drawsocket JSON format and extends it by specifying *expressions* associating Picster graphics with code to be executed in the Max environment [5]. In *Picster Button Mode*, a Picster Element functions as an interactive button sending its expression on a mouse click or touch event. This also works in external devices by communicating mouse or touch events over a wide-area or local network.

3. REALIZATION

3.1. Score Generation

The aim was to recreate Erdmann's score by turning every measure and box into an individual Picster element and to attach expressions for real-time user interaction in button mode. For this, the score was sliced into separate bitmap images, traced as vector graphic using suitable applications such as Adobe Illustrator or Inkscape and saved in SVG format.

In MaxScore, three documents corresponding to the pages of the original score were created with the number of measures equal to the number of elements in the score. The SVG images were dragged and dropped onto the selected measures, thus importing them as Picster Elements. Proper spacing between measures was achieved by using the MaxScore *setMeasureLeftMargin* and *setMeasureWidth* messages so that an image could be mapped to the width of a measure and distances between playable areas of individual measures could be set. Once attached to measures, the resulting Picster Elements were edited to change their size or color or to remove artefacts.

The colors used in the score were: magenta = violin; blue = cello; green = double bass; orange = percussion; black = tutti; red = sound files only.

3.2. Cursors

It was decided that measures containing music notation as well as some of the boxes would feature cursors traversing the horizontal distance according at the speed calculated from the tempo and time signature settings for each measure (see Figure 1). Cursors can be created and executed in MaxScore with messages such as

```
cursor 0 @begin 7 0 @end 7 0 @timestretch 1 @passes 1 @color 1 0 1 1
```

or

```
cursor 0 stop.
```

These messages were subsequently attached to Picster elements as expressions to be executed during the performance. Since MaxScore allows up to 20 cursors to move simultaneously, cursor instance numbers were chosen according to the instruments they represent. The cursor *timestretch* attribute allows multiple cursors to traverse the same measure at different speeds.

Les Sons Visionnaires p. 1

Line	Bar	Instrument	Tempo
1	1	Bass	40
1	2	Drum kit	34
1	3	Cello	40
1	4	Drum kit	20
1	5	Violin	40
1	6	Bass	48
2	1	Tutti	40
2	2	Violin Cello Bass	40
2	3	Electronics	44
2	4	Drum kit	30
3	1	Tutti	30
3	2	Violin	40
3	3	Cello	30
4	1	Electronics	17
4	2	Violin	22
4	3	Tutti	16

Les Sons Visionnaires p. 2

Line	Bar	Instrument	Tempo
1	1	Cello	68
1	2	Drum kit	22
1	3	Bass	27
1	4	Drum kit	60
2	1	Drum kit	22
2	2	Drum kit	22
2	3	Tutti	53
2	4	Violin Cello Bass	32
2	5	Cello	60
2	6	Electronics	96
3	1	Violin	34
3	2	Drum kit	32
3	3	Tutti	68
4	1	Bass	32
4	2	Violin	32
4	3	Bass	32
4	4	Drum kit	32
4	5	Violin	78
5	1	Cello	28
5	2	Drum kit	52
5	3	Tutti	76
5	4	Tutti	20
5	5	Violin Cello Bass	56
5	6	Drum kit	34
6	1	Tutti	36
6	2	Violin	88
6	3	Cello	40
6	4	Violin Cello Bass	64
6	5	Electronics	60
7	1	Violin Cello Bass	40
7	2	Tutti	20
7	3	Bass	66
7	4	Electronics	24
7	5	Violin Cello Bass	40

Les Sons Visionnaires p. 3

Line	Bar	Instrument	Tempo
1	1	Cello	60
1	2	Tutti	32
1	3	Violin	60
2	1	Tutti	38
2	2	Violin Cello Bass	48
2	3	Drum kit	44
3	1	Cello	90
3	2	Drum kit	53
3	3	Violin Cello Bass	60
3	4	Drum kit	60
3	5	Electronics	34
3	6	Violin	60
4	1	Bass	53
4	2	Violin	53
4	3	Bass	60
4	4	Electronics	96
4	5	Tutti	58
5	1	Violin	45
5	2	Drum kit	48
5	3	Cello	28
6	1	Tutti	40
6	2	Bass	30

1. bars color according to instrument
2. is it possible to make cursors to start a bit later after click? 1.2.2

Figure 1. Cursors placements were specified by the conductor of the ensemble

3.3. Sound Files

The boxed elements were interpreted sonically by the participants of the computer-assisted composition class yielding a total of 68 sound files. We devised a Max patch facilitating simultaneous playback of up to 4 sound files by sending messages in the format

`sf <path_to_sound_file>.`

These messages were attached to the corresponding Picster elements via the Picster sf expression editor—an integral part of the MaxScore Editor (Figure 2). Just as with cursor actions, sound file playback can be achieved by touching the boxes displayed on the tablet computers. Some of the boxes contain multiple expressions for both cursor and sound-file playback messages.

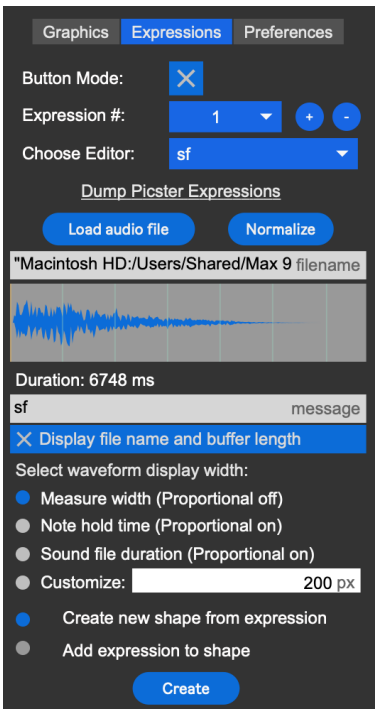


Figure 2. The sound file expression editor is part of the MaxScore Editor and was used to assign sound files to Picster Elements

3.4. Network Setup

In the performance, Apple iPads were connected wirelessly via a dedicated router to the main computer running the Max patch with the Drawsocket web server. The three pages, stored in separate Max dictionaries, were dynamically displayed by clicking on the page labels displayed on top of the score. During the performance, the conductor created a non-linear dramaturgy by jumping from page to page, combining similar or contrasting elements which he "fired up" by touching the elements of choice. This would make the objects flash on all connected tablets, thus drawing the attention of the performers to the new event. The elements were then supposed to be executed/interpreted by a player or a group of players provided the cursor color match the color assigned to them (Figure 3).

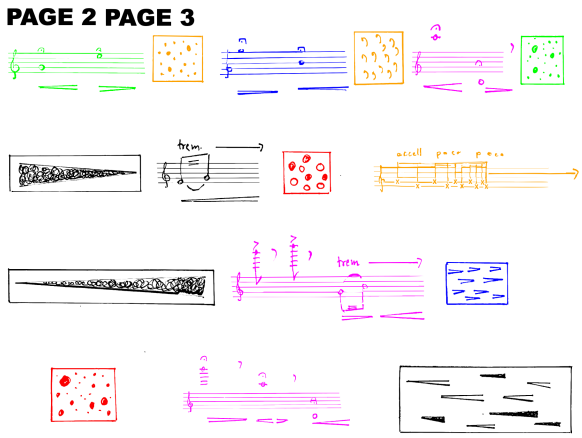


Figure 3. Page 1 of the score as it appears on the participants' tablet computers.

4. THE CONCEPT OF THE INTERACTIVE SCORE

Interactive scores of the kind we realized, serve a dual purpose: as both display and instrument. The conductor must learn the affordances of the score as if he was playing an instrument. It thus combines and extends the traditional roles of the conductor and the performer. Because of its networked nature and simplicity of interaction, it is also an excellent tool for children who can be introduced to contemporary digital music practices.

5. TENOR 2025 PERFORMANCE

We are proposing a performance of *Les Son Visionnaires* during the 2025 Beijing TENOR conference. Four scenarios are imaginable, and we are requesting feedback on their feasibility.

5.1. Conductor-centric local network performance

The conductor retains control of the performance and oversees its dramaturgy.

5.2. Performer-centric local network performance

The performers make the decisions on their own actions and those of their fellow performers. Decisions can be overwritten by peers.

5.3. Participatory local network performance

The audience or select members thereof are given tablets with the instruction as to how to guide the performance. The participants receive a brief introduction to the piece so they can familiarize themselves with the affordances of the system.

5.4. Wide-area network performance

The conductor and performers play from remote locations connecting to an instance of the Max patch running on a server at the *ligeti center* in Hamburg. Due to the nature of

this performance scenario, control over its dramaturgy should remain with the conductor.

6. INSTRUMENTATION

We are proposing the following instrumentation:

- Flute
- Banhu
- Guzheng
- Percussion

The performers should be physically distributed either inside or outside the performance venue (see section 6)

7. TECHNICAL REQUIREMENT

A local network performance would require the following equipment:

- MacBook Pro (preferred) or PC running Max, MaxScore and D
- 5 iPads or tablet computers with built-in browsers
- A dedicated 5 GHz router
- Multichannel playback (scalable from 2 to 8 or more speakers)
- Projection of the main computer screen

In addition, a wide-area performance would require:

- Access to the Internet with port 3002 opened for websocket connection and VNC enabled for remote control of the server
- Low-latency audio connection via JackTrip or similar
- Video-streaming via Zoom

8. LINK TO SUPPORTING MATERIALS

Please find a 00:42 video clip as well as a 08:41 audio recording of two different performance of *Les Son Visionnaires* at this link: https://tinyurl.com/4rudbrck

Acknowledgments

I'd like to acknowledge the contributions of Luki Becker, Oscar Corpo, Agustín Issidoro, Christopher Ramm, Candid Rütter and Hyewon Son in the realization of *Les Son Visionnaires* by Helmut W. Erdmann.

9. REFERENCES

1. Hajdu, G. and Didkovsky, N.: MaxScore – Current State of the Art. Proceedings of the International Computer Music Conference. (2012)
2. Didkovsky, N. and Hajdu, G.: MaxScore Music Notation in Max/MSP. Proceedings of the International Computer Music Conference. (2008).
3. Gottfried, Rama, and Georg Hajdu. Drawsocket: A Browser Based System For Networked Score Display. Proceedings of the Fifth International Conference on Technologies for Music Notation and Representation, Melbourne (2019).
4. Hajdu, Georg, and Rama Gottfried. Networked Music Performance In The St. Pauli Elbe Tunnel. Proceedings of the Fifth International Conference on Technologies for Music Notation and Representation, Melbourne (2019).
5. Hajdu, G. and Didkovsky, N.: MaxScore: Recent Developments. Proceedings of the Fourth International Conference on Technologies for Music Notation and Representation, Montréal (2018).