

Rethinking the audio workstation : tree-based sequencing with i-score and the LibAudioStream

Jean-Michel Celerier

Affiliation1

author1@smcnetwork.org

Myriam Desainte-Catherine

Affiliation2

author2@smcnetwork.org

Stphane Letz

Affiliation3

author3@smcnetwork.org

ABSTRACT

Place your abstract at the top left column on the first page. Please write about 150–200 words that specifically highlight the purpose of your work, its context, and provide a brief synopsis of your results. Avoid equations in this part.

1. INTRODUCTION

Progression du papier -> poser le problème -> définitions et présentation des outils -> extensions de la libaudiostream -> traduction de i-score en expression libaudiostream

2. EXISTING WORKS

- Audiographe, etc. : Jamoma Audiograph, Logic - Ableton Live - Reaper

3. CONTEXT

In this section, we will present the two tools that are used to achieve rich audio sequencing : i-score and the LibAudioStream. i-score is an interactive sequencer which allows to position events in time, and gives the possibility to introduce interaction points and conditions in the score. The detailed execution semantics are given in [?].

The LibAudioStream provides the ability to write rich audio expression by creating and combining streams. The introduction of symbolic date, introduced in allows to start and stop the execution of streams at an arbitrary date.

3.1 Description i-score

- interactivité -> mapping and JS -> donner exemple scénario i-score

3.2 Description LibAudioStream

-> donner sémantique de flux. -> donner exemple flux stream

4. PRESENTATION OF THE SYSTEM

4.1 Group audio stream

4.2 Send and return audio streams

- Flux infinis

4.3 Stream graph

-> donner sémantique de flux des Stream Group, Send, Return.

- Comme la durée de chaque contrainte peut varier avec le ralentissement, on utilise principalement des dates symboliques -> Processus audio dans i-score : - FX -> Supports FaUST. - Instrument. - Send. - Return. - Mixing. - Hiérarchie profondeur arbitraire. - Automations : exporte les paramètres dans le modèle d'objet

Faire graphe pour une Time Constraint et donner un exemple avec effets appliqués sur scénario. Expliquer graphe hiérarchique de dépendances : penser au cas où un a un return dans une hiérarchie puis un send un niveau supérieur; il faut faire le graphe de A Z et s'assurer qu'il ne soit pas cyclique

1er cas : Un son avec une piste d'effets.

2ème cas : scénario hiérarchique, boucle

Cas de la boucle avec un coup A, un coup B selon la condition ? -> exécution d'un timenode doit reset le flux.

Piste send / return : permet de maintenir les queues de reverb.

4.4 Routing, multi-channels, etc.

-> mettre maquettes track mix

5. EXAMPLE

5.1 UI

-> capture d'écran Faire vue scénario et sa traduction en graphe de routage

6. CONCLUSION

-> lackluster areas : - MIDI support (but OSC) - no musical time information : first aimed for artists, but improvements could be the waiting of triggering on some measure of time. - "play from anywhere" - audio input ? - correction de latence ?

Acknowledgments

ANRT, Blue Yeti, Magali

7. REFERENCES

- [1] A. Someone, B. Someone, and C. Someone, “The title of the conf. paper,” in *Proc. Int. Conf. Sound and Music Computing*, Porto, 2009, pp. 213–218.
- [2] X. Someone and Y. Someone, *The Title of the Book*. Springer-Verlag, 2010.
- [3] A. Someone, B. Someone, and C. Someone, “The title of the journal paper,” in *J. New Music Research*, 2008, pp. 111–222.