

Compte-rendu à t=2.0y

Jean-Michaël Celerier

November 10, 2016

Contents

1	Introduction	5
1.1	Mise en relation avec le sujet	5
1.2	Articulation et analyse générale	5
2	Réalisations	7
2.1	Développements théoriques, publications	7
2.1.1	États de l'art	7
2.1.2	Modèle théorique	7
2.1.3	Espace	7
2.1.4	Audio	8
2.1.5	Répartition	8
2.2	Conférences, présentations, workshops	8
2.2.1	Cycles SCRIME 2015	8
2.2.2	Forum IRCAM 2015	8
2.2.3	FOSDEM 2016	8
2.2.4	Cycles SCRIME 2016	8
2.2.5	GDR ESARS	8
2.2.6	DESINC2016	8
2.2.7	Workshop improvisation	8
2.3	Développements logiciels	8
2.3.1	Génie logiciel et généralités	8
2.3.2	i-score	8
2.3.3	Extensions à i-score	9
2.3.4	libossia	9
2.3.5	OSCQuery	9
2.3.6	coppa	9
2.3.7	Études et développements mineurs	9
2.4	Projets liés	10
2.4.1	Audio	10
2.4.2	Robots	10
2.5	Cours et TDs donnés	11
2.5.1	TIM	11
2.5.2	TAP	11
3	Objectifs à venir	13
3.1	Système réparti	13
3.1.1	Exécution répartie	13
3.1.2	Répartition des protocoles	13

3.2	Audio	13
3.2.1	Article dans CMJ ?	13
3.2.2	Signatures temporelles	13
3.2.3	Support audio étendu	13
3.3	Embedding de i-score	14
3.3.1	DLL dans d'autres moteurs d'exécution	14
3.3.2	Scénarios compilés	14
3.3.3	Web	14
3.3.4	IncludeOS pour devices?	14
3.4	Unification temps - espace	14
3.5	Modèle par graphe de noeuds pour calcul par tranches	14
3.6	Objectifs personnels	14
4	Conclusion	15

Chapter 1

Introduction

Ce document survole les travaux qui ont été réalisés jusqu'à présent lors de la thèse, étudie les pistes qui sont ouvertes et les possibilités pour la dernière année.

1.1 Mise en relation avec le sujet

Calques audio interactifs : théorie, mise en oeuvre et usages.

1.2 Articulation et analyse générale

Chapter 2

Réalisations

2.1 Développements théoriques, publications

2.1.1 États de l'art

2.1.2 Modèle théorique

TENOR2015: OSSIA

IUI2015 (refusé)

JNMR: Vérification

JIM2016: Interface

ICMC2016: Programmation structurée

2.1.3 Espace

JIM2016: Démo

JIM2016: Espace

-> Conclusion : CAS peu adéquat, dur d'avoir de bonnes performances à un tick rate quelconque. Alternatives : se restreindre aux cas linéaires ? GPU ? Mais latence.

Compte-rendu espace

2.1.4 Audio

SMC2016: i-score et LibAudioStream

2.1.5 Répartition

Rapport de stage

2.2 Conférences, présentations, workshops

2.2.1 Cycles SCRIME 2015

2.2.2 Forum IRCAM 2015

2.2.3 FOSDEM 2016

2.2.4 Cycles SCRIME 2016

2.2.5 GDR ESARS

2.2.6 DESINC2016

2.2.7 Workshop improvisation

2.3 Développements logiciels

2.3.1 Génie logiciel et généralités

Performances

Question des performances ? Comment mettre en valeur ? Un accent très fort est mis dessus.

Tests

Idem pour tests. Couverture de code : 70 % pour libossia, 50 % pour i-score, 0 % pour extensions i-score

2.3.2 i-score

Architecture

Problèmes actuels

Portabilité

Après avoir enlevé Jamoma, exécution sur Android et iOS.

2.3.3 Extensions à i-score

Édition répartie

Audio

Automation 3D

PureData

Espace

Image

Vidéo

Contrôle à distance

Analyse statique

Segments

Extension "Preset"

2.3.4 libossia

Architecture

Problèmes actuels

- temps de compilation

Portages

C

Csharp et Unity

Qt

Java

Javascript

2.3.5 OSCQuery

2.3.6 coppa

2.3.7 Études et développements mineurs

External RealSense

Outils pour graphe de calcul

DisPATCH

RaftLib

Contribution à d'autres projets open-source**LibAudioStream****FAUST****Jamoma****Contributions mineures**

- Placeholder/Nodeeditor
- verdigris
- fnt
- Qt-color-widgets
- jni.hpp
- quazip
- QRecentFilesMenu
- ModernMIDI
- libsamplerate
- ofxMSAPhysics
- Cotire

2.4 Projets liés**2.4.1 Audio****Stage Magali Chauvat****Objectifs****2.4.2 Robots****Stage Nicolas 2015****Stage Kinda Al Chahid 2015****Stage Paul Breton 2016****Stage Maëva 2016****Projet TM - Robot 2015 - 2016****Objectifs****Groupe TM**

Groupe Robots

Projet TM - Robot 2016 - 2017

Objectifs

Groupe TM

Groupe Robots

PFA2016 - 2017

Objectifs

2.5 Cours et TDs donnés

2.5.1 TIM

2.5.2 TAP

Chapter 3

Objectifs à venir

3.1 Système réparti

3.1.1 Exécution répartie

3.1.2 Répartition des protocoles

3.2 Audio

3.2.1 Article dans CMJ ?

Pour que ce soit convainquant : offrir en plus la possibilité de réutiliser les flux passés. Et bien tout modéliser.

3.2.2 Signatures temporelles

3.2.3 Support audio étendu

VST / VSTi

LV2

Format de plug-ins qui permet l'analyse en temps réel de données.

3.3 Embedding de i-score

3.3.1 DLL dans d'autres moteurs d'exécution

3.3.2 Scénarios compilés

3.3.3 Web

3.3.4 IncludeOS pour devices?

3.4 Unification temps - espace

- Lister les cas possibles : comprendre ce que "appliquer les structures de i-score dans l'espace" veut dire. Ex : - si on considère qu'une contrainte temporelle

est toujours sur une seule dimension, de temps ? - si on considère qu'une contrainte temporelle est toujours sur une seule dimension, d'espace ? - si on considère qu'une contrainte temporelle est sur plusieurs dimensions d'espace ? - si on considère qu'une contrainte temporelle est sur toutes les dimensions à la fois (on fait "progresser" l'espace temps et on arrive dans des espaces différents selon les outcome du scénario i-score). - si on considère non pas l'avancement dans un scénario comme quelque chose de global, mais quelque chose de lié à un acteur. - on a des lors besoin de multiples curseurs de temps.

3.5 Modèle par graphe de noeuds pour calcul par tranches

3.6 Objectifs personnels

Chapter 4

Conclusion