

# DES SIGNAUX AUX SYMPHONIES: POUR UNE MODÉLISATION HOMOGENÈME DES OBJETS SONORES



David Janin & Myriam Desainte-Catherine  
JIM 2015  
Montréal, Québec, Canada

# Plan

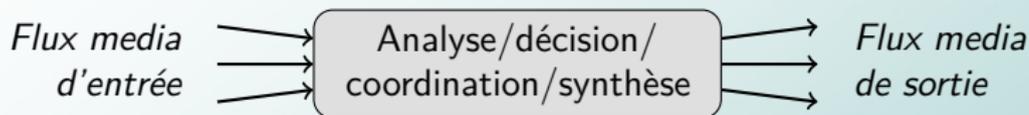
- Problématique générale :  
Simplifier l'accès aux outils de CAO
- De la conception à la mise en oeuvre :  
Conception hors temps vs execution dans le temps ?
- Vers un noyau d'outils uniforme :  
Programmation et modélisation par tuilage
- Exemples d'applications :  
Du signal à la symphonie (in memoriam Paul Hudak)

# Plan

- **Problématique générale :**  
Simplifier l'accès aux outils de MAO
- De la conception à la mise en oeuvre :  
Conception hors temps vs execution dans le temps ?
- Vers un noyau d'outils uniforme :  
Programmation et modélisation par tuilage
- Exemples d'applications :  
Du signal à la symphonie

# Domaine de recherche

## Design d'application media interactive

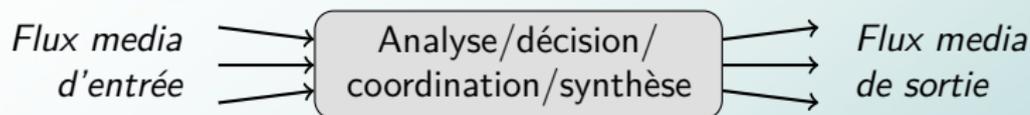


## Caractéristique principale de ces applications

- Omniprésent (art, communication, assistance, ...)
- Hétérogène (son, video, contrôle, gestes, ...)
- Multi-disciplinaire (traitement du signal, programmation, ...)
- Nombreux outils (capture, analyse, transform., product., ...)
- Designers créatifs non informaticiens (cas d'utilisation spécifiques, métaphores de design adhoc)

avec un besoin croissant de **productivité**, de **robustesse** et de **fiabilité**.

# Objectifs de recherche



Développer des **modèles**, des **méthodes** et des **outils** de design qui soient :

- Simples (les artistes/débutants peuvent les utiliser),
- Efficaces (les flux se combinent et s'exécutent bien),
- Robustes (applicables à la production d'applications réelles),

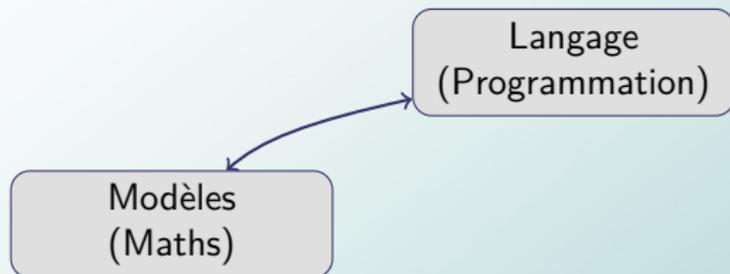
et aussi

- Homogènes (transversals aux outils existants),
- Modulaires (pour un design incrémental et hiérarchique),
- Cohérents (avec de bonnes propriétés mathématiques).

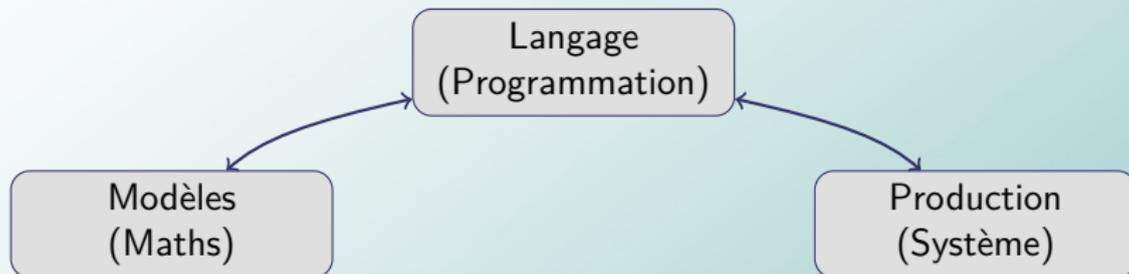
# Approche orientée modèle

Modèles  
(Maths)

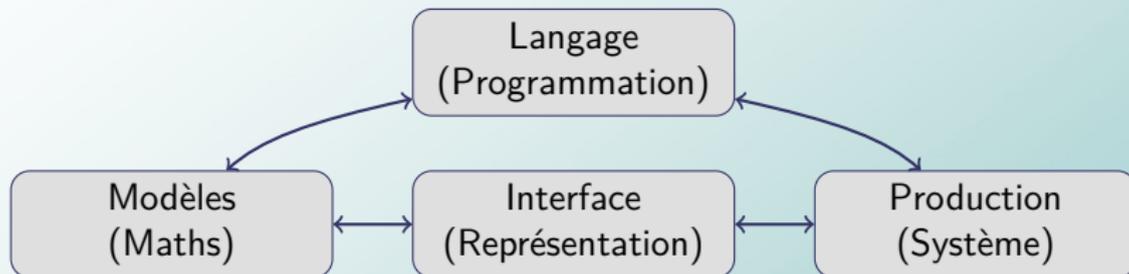
# Approche orientée modèle



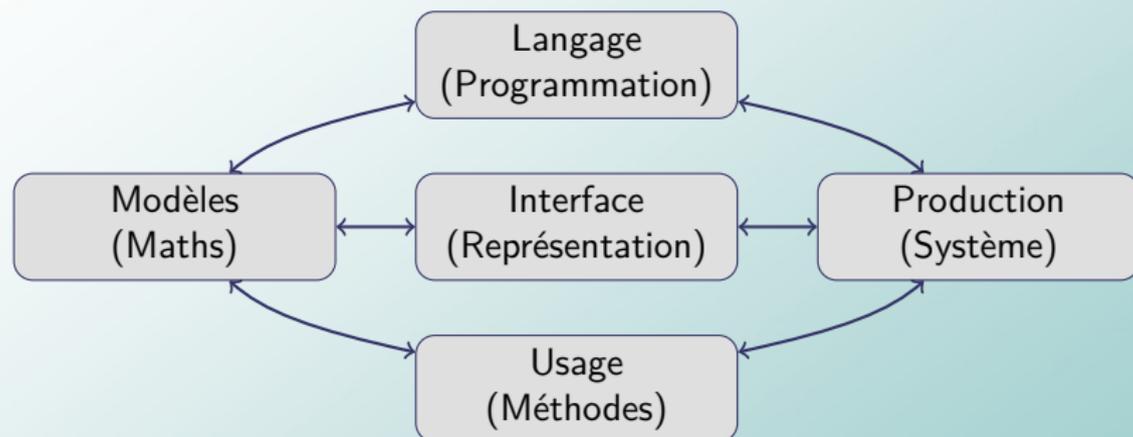
# Approche orientée modèle



# Approche orientée modèle



# Approche orientée modèle



# Plan

- Problématique générale :  
Simplifier l'accès aux outils de MAO
- De la conception à la mise en oeuvre :  
Conception hors temps vs execution dans le temps ?
- Vers un noyau d'outils uniforme :  
Programmation et modélisation par tuilage
- Exemples d'applications :  
Du signal à la symphonie

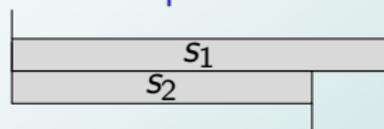
# Modélisation dans le temps

## Le temps de la performance

### Produit séquentiel



### Produit parallèle



### Est-ce suffisant ?

- Pour l'informaticien : oui!
- Pour le musicien : faut voir...

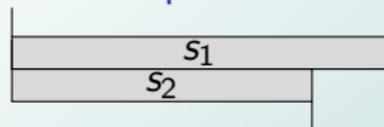
# Modélisation dans le temps

## Le temps de la performance

### Produit séquentiel



### Produit parallèle



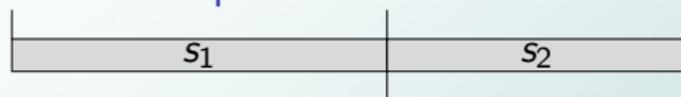
### Est-ce suffisant ?

- Pour l'informaticien : oui !
- Pour le musicien : faut voir...

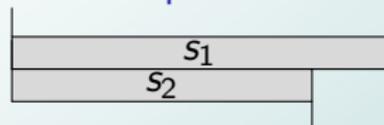
# Modélisation dans le temps

## Le temps de la performance

### Produit séquentiel



### Produit parallèle



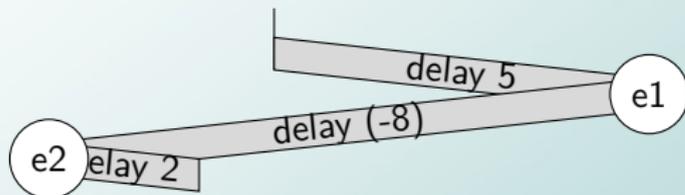
### Est-ce suffisant ?

- Pour l'informaticien : oui !
- Pour le musicien : faut voir...

# Modélisation hors temps

Le temps de la composition

Zigzags hors temps

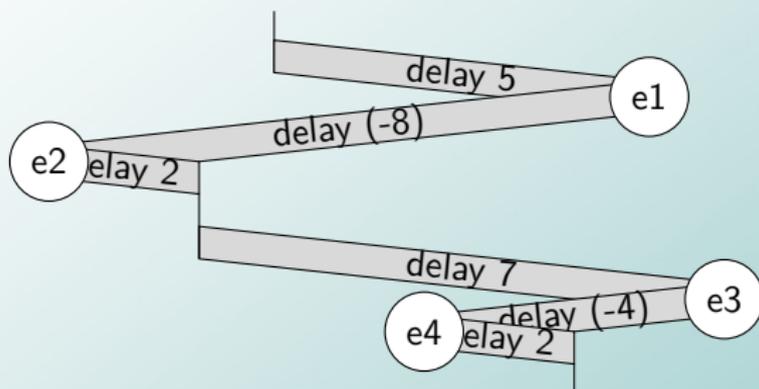


Ordonnées pour une exécution dans le temps

# Modélisation hors temps

Le temps de la composition

## Zigzags hors temps

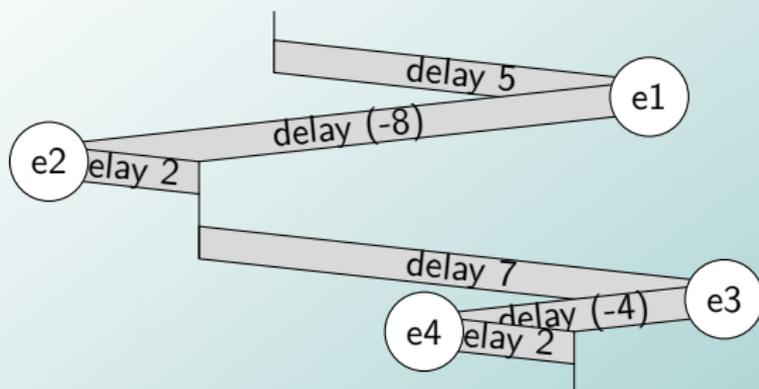


Ordonnances pour une execution dans le temps

# Modélisation hors temps

Le temps de la composition

## Zigzags hors temps



## Ordonnés pour une execution dans le temps

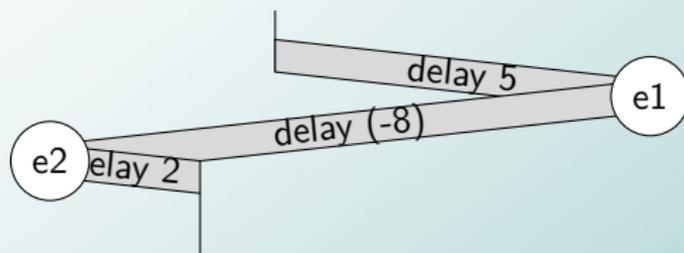


# Plan

- Problématique générale :  
Simplifier l'accès aux outils de MAO
- De la conception à la mise en oeuvre :  
Conception hors temps vs execution dans le temps ?
- Vers un noyau d'outils uniforme :  
Programmation et modélisation par tuilage
- Exemples d'applications :  
Du signal à la symphonie

# Objets mathématiques sous-jacents

## Flux et produits tuilés induits



Produit tuilé = synchronisation + fusion

# Objets mathématiques sous-jacents

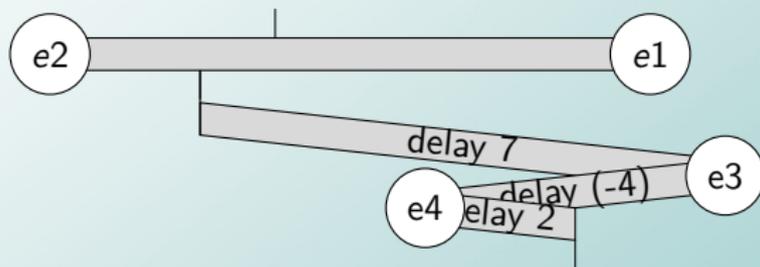
## Flux et produits tuilés induits



Produit tuilé = synchronisation + fusion

# Objets mathématiques sous-jacents

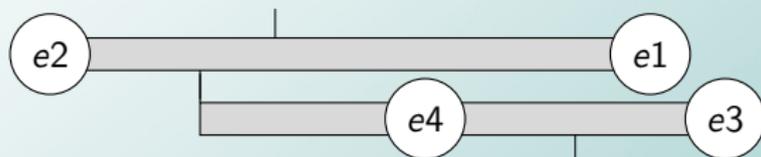
## Flux et produits tuilés induits



Produit tuilé = synchronisation + fusion

# Objets mathématiques sous-jacents

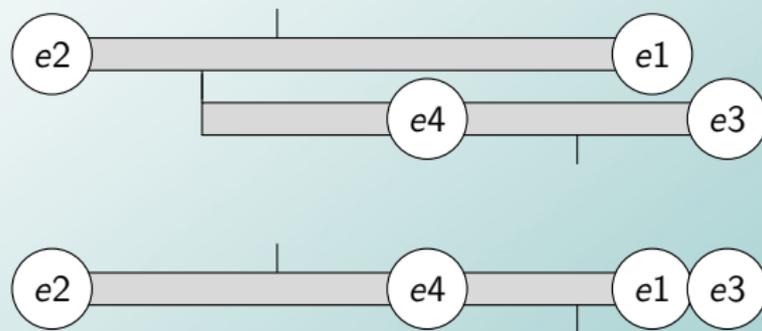
## Flux et produits tuilés induits



Produit tuilé = synchronisation + fusion

# Objets mathématiques sous-jacents

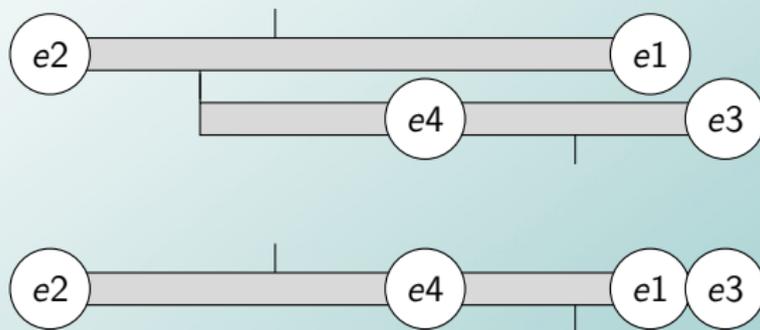
## Flux et produits tuilés induits



Produit tuilé = synchronisation + fusion

# Objets mathématiques sous-jacents

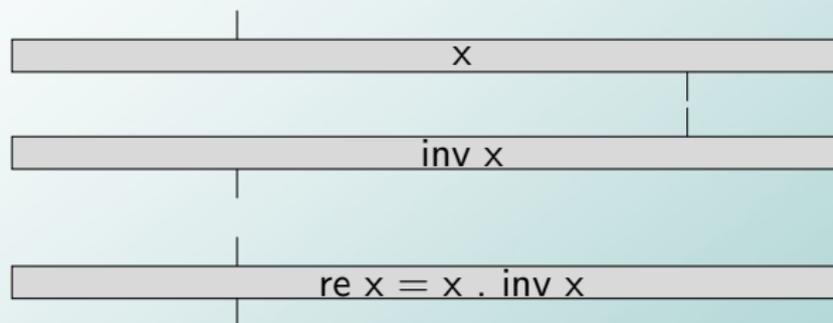
## Flux et produits tuilés induits



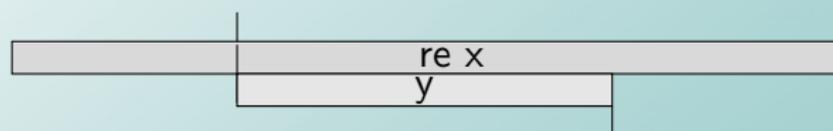
Produit tuilé = synchronisation + fusion

# Opération dérivée : reset et pré-synchronisation (Fork)

Reset

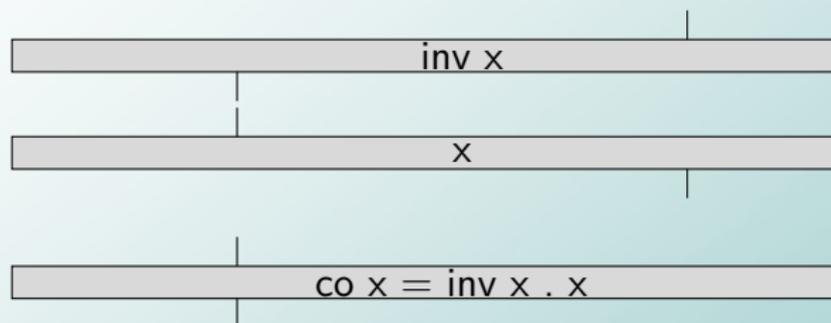


Pré-synchronisation :  $\text{fork } x \ y = \text{re } x \ . \ y$

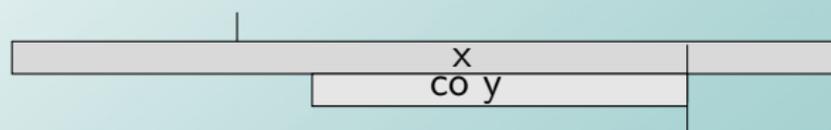


# Opération dérivée : co-reset et post-synchronisation (Join)

Co-reset

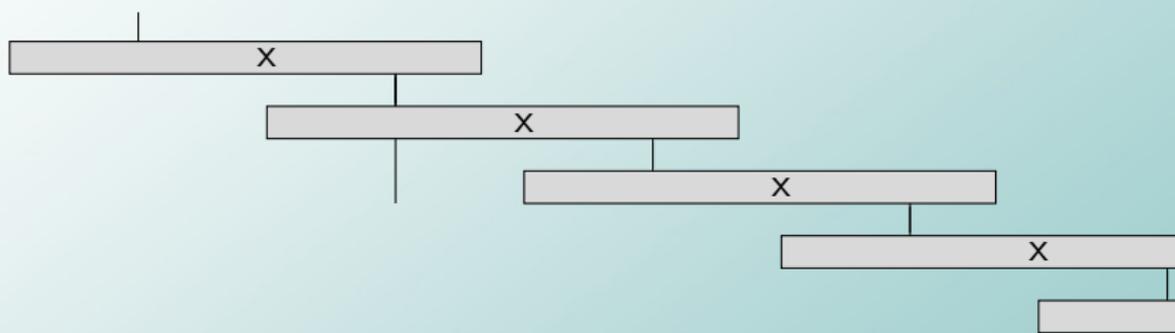


Post-synchronisation :  $join\ x\ y = x . co\ y$



# Flux infini (paresseux)

Equation de point fixe :  $y = x . \text{re } y$



# Implementation

En Haskell

## Bibliothèque en cours de développement (S. Archipoff)

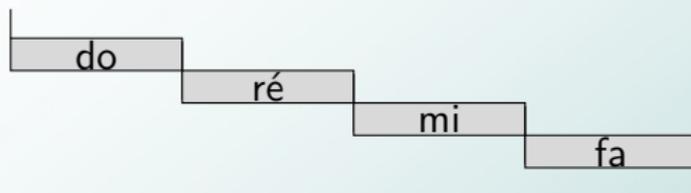
- version évènementielle (event + delay) avec multiset d'évènements,
- version audio (value + duration) avec multiset de valeurs,
- implémentation efficace en Skew list ( $\log n$ ),
- etc. . .

# Plan

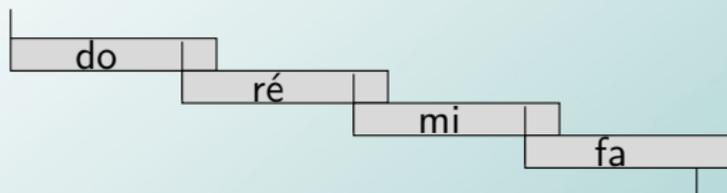
- Problématique générale :  
Simplifier l'accès aux outils de MAO
- De la conception à la mise en oeuvre :  
Conception hors temps vs execution dans le temps ?
- Vers un noyau d'outils uniforme :  
Programmation et modélisation par tuilage
- Exemples d'applications :  
Du signal à la symphonie

# Tuiles MIDI : de la partition à l'interprétation

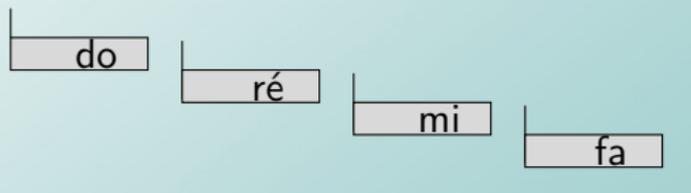
## Partition



## Legato

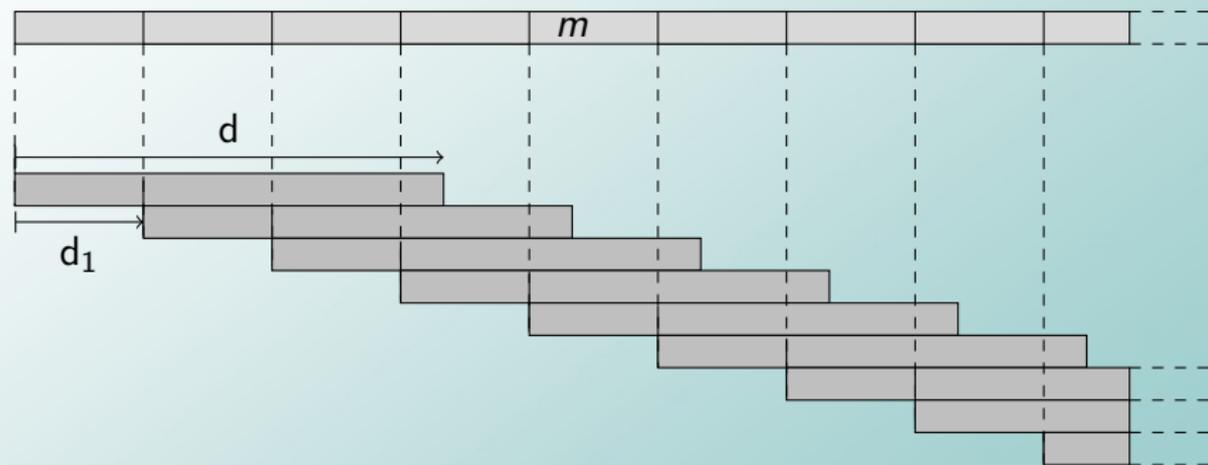


## Pizzicato



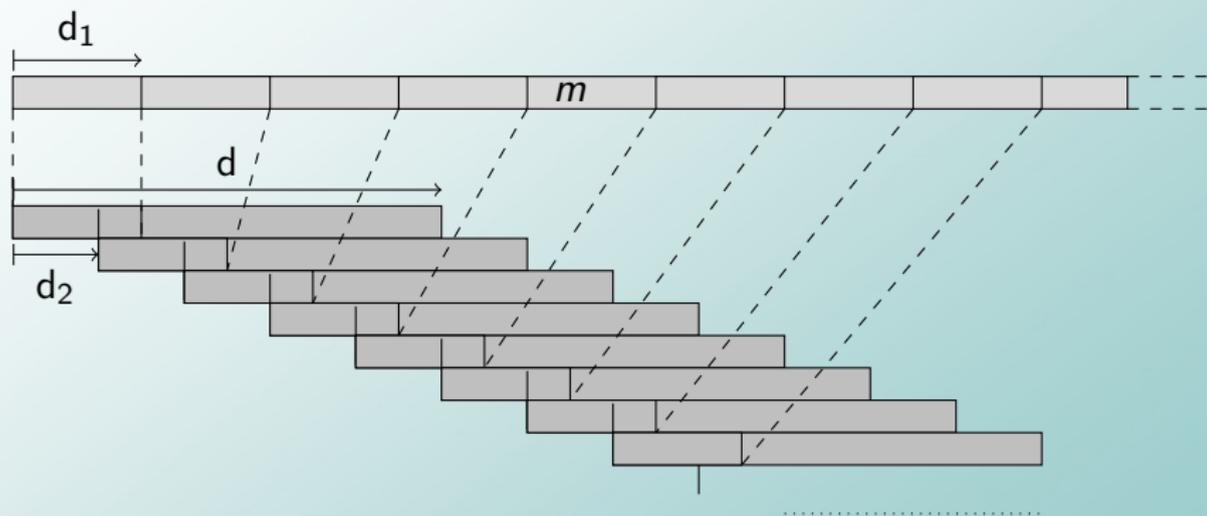
# Tuile audio : technique granulaire

## Décomposition/recomposition granulaire



# Tuile audio : technique granulaire

## Décomposition/contraction/recomposition granulaire



# Conclusion

Un projet au long terme

## Ce qui est fait

- théorie des tuiles linéaires,
- codage efficace en Haskell,
- scheduler/player audio et MIDI,
- théorie des tuiles de dimension supérieure (graphes).

## Ce qui reste à faire

- expérimentations musicales,
- bibliothèque de fonctions orientées utilisateurs,
- expérimentation modélisation hiérarchique/multi-échelle,
- intégration théorique et codage d'évènements non contrôlables (Input),
- codage tuiles de dimension supérieure (graphes),
- représentation/édition graphique des programmes tuilés.

# Conclusion

Un projet au long terme

## Ce qui est fait

- théorie des tuiles linéaires,
- codage efficace en Haskell,
- scheduler/player audio et MIDI,
- théorie des tuiles de dimension supérieure (graphes).

## Ce qui reste à faire

- expérimentations musicales,
- bibliothèque de fonctions orientées utilisateurs,
- expérimentation modélisation hiérarchique/multi-échelle,
- intégration théorique et codage d'évènements non contrôlables (Input),
- codage tuiles de dimension supérieure (graphes),
- représentation/édition graphique des programmes tuilés. ▶

Merci pour votre attention

# 3rd CFP FARM 2015

3rd ACM SIGPLAN International Workshop on  
Functional Art, Music, Modelling and Design

Vancouver, Canada, 5 September, 2015

affiliated with ICFP 2015

Full Paper and Demo Abstract submission

Deadline : 17 May