

# Outils spatiaux pour partitions interactives

Jean-Michaël Celerier  
Myriam Desainte-Catherine  
Jean-Michel Couturier

1er avril 2016

Mais d'abord, qu'est-ce qu'un espace ?

→ Un ensemble de dimensions bornées continues.

# Problématique

- ▶ Comment définir des scénarios interactifs contenant des éléments spatiaux.
- ▶ Comment gérer des comportements multidimensionnels dans le temps.
- ▶ ()UI d'écriture.

## Objectif :

- ▶ Scénariser des interactions à plus d'une dimension.
- ▶ Exemple : **muséographie, contrôle de robots**, jeux vidéos, musique spatialisée...

# Approche

- ▶ Approche **déclarative**.
- ▶ Généralisation puis réduction :
- ▶ Implémentation dans cas générique : forme cartésienne calculée par un CAS.
- ▶ Implémentations optimisées dans cas courants (OO).
- ▶ Alternative : Constructive Solid Geometry.

## Définition (Espace)

- ▶ On crée d'abord un espace : nombre de dimensions, bornes, granularité.

Ex. :

- ▶  $x \in [-100; 100]$
- ▶  $y \in [0; 50]$
- ▶ Granularité : 5.

## Définition (Zone)

- ▶ On définit des zones en donnant un ensemble d'**équations cartésiennes** qui les définissent.
- ▶ On sépare les variables en deux groupes : variables d'espace et paramètres.
- ▶ **Transformations** : échelle, rotation, translation.
- ▶ Fonctions autorisées : admissibles dans solutions en **forme close** ( $\sin x$ ,  $\cosh x$ ,  $\ln x$ ,  $\sqrt{x}$ , ...). Pas d' $\int$ .
- ▶ Paramètres des zones : peuvent être des constantes ou bien des adresses OSC.

# Définition (Zone)

Ex. :

$$u \leq 0 \ ; \ (u - x_0)^2 + (v - y_0)^2 \leq r_0^2$$

- ▶  $u \leftarrow x$
- ▶  $v \leftarrow y$
- ▶  $x_0 \leftarrow \text{foo} : / \text{bar}$
- ▶  $r_0 \leftarrow \text{parent} : / \text{t}$
- ▶  $y_0 \leftarrow 5$

## Définition (Calculs et exécution)

- ▶ On définit des calculs et relations qui vont être évaluées entre zones.
- ▶ Actuellement : **collision**, **distance** (des barycentres).
- ▶ À terme : pouvoir là aussi spécifier des formules (nécessite l'accès aux variables des zones).



# Définition (Exécution)

Sémantique d'exécution :

- ▶ On suit celle d'i-score : au tick.
- ▶ **Utilisation récursive** des résultats : décalage d'un tick.

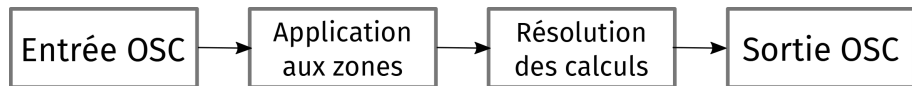


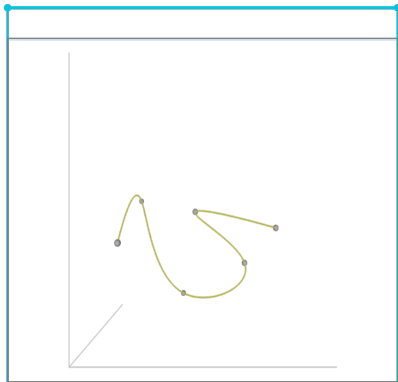
FIGURE – Flot de calcul à l'exécution

# Visualisation

- ▶ Actuellement, seulement 2D.
- ▶ Passage à la 3D : **VTK** ? **Qt3D** ? Pour zones génériques : triangulation (Delaunay).
- ▶ Passage en  $n$  dim : par restriction à des sous espaces.
- ▶ À faire sur GPU pour cas général.

# Automations 3D

- ▶ Standard : équation paramétrique. Comme **IanniX**, **OpenMusic**...
- ▶ Exécution : courbe parcourue dans son intégralité durant une contrainte temporelle.



# Démonstration

# Conclusion

- ▶ Présentation d'un modèle de définition d'**objets spatiaux** dans i-score.
- ▶ Géométries **dynamiques**.
- ▶ Splines paramétriques traditionnelles sont présentes aussi.
- ▶ **Problème** : représentation disjointe d'un seul espace ?
- ▶ **Problème** : représentation plus propre des ouverts ?
- ▶ **Problème** : Écriture de scénarios spatiaux contraints : analogue d'i-score dans plus que 1D.

# Perspectives

- ▶ Généraliser les deux en une méthode permettant d'étendre la notion de scénario interactif à des flux multi-dimensionnels. → Actuellement, le lien est fait à la main dans i-score.
- ▶ Notion d'abstraction dans i-score : patch ?
- ▶ Compiler à la volée pour exec. plus rapide ? (Actuellement : `vtkFunctionParser`).

Merci de votre attention !

# Questions