

Outils d'Edition d'Animation d'Humanoïde de Synthèse.

To Do :

Motion Blending Editor.

Voici globalement l'ensemble des fonctionnalités qu'il restent à implémenter au sein du Motion Blending Editor, pour qu'il deviennent un outil très efficace. L'ensemble de ces Fonctionnalités ne sont pas vitales, mais néanmoins feraient du Motion Blending Editor, un outil extrêmement puissant et fiable. Ces propositions ne sont ni exhaustives, ni absolues, mais restent néanmoins des pistes que nous estimons intéressantes à développer, et dans l'intérêt de l'utilisateur final.

Le code du Motion Blending Editor est dans sa structure beaucoup plus accessible que celui du ConstraintEditor (moins de classes). Vous serez néanmoins guidés dans les grandes lignes de notre implémentation par la documentation du code.

Implémentations nécessaires :

Undo/Redo :

- Gestion des manipulations du graphe sur le canvas, et des manipulations de propriété de blend. Le stockage

Chargement :

- Importation de tous les fichiers d'un dossier.
- Ouverture d'un projet Motion Blending Editor.

Sauvegarde:

- Sauvegarde du projet, et de son contexte associé (fichiers, position des Items sur le canvas..)

Canvas Controller:

- Implémenter les fonctions copier/couper/coller. Lors de ces opérations, tenir compte des liens uniquement entre chaque Item faisant partie de la sélection, et ce de façon cohérente. (ça n'a pas d'intérêt de copier le lien d'un seul des parents d'un blend).
- L'implémentation de fonction de serialisation/deserialisation utilisée par hLib/AReVi est un bon exemple d'introspection et de reification de classes.

Il serait très pratique à utiliser pour l'implémentation de Undo/Redo, mais aussi celle des Chargements/Sauvegarde.

Propositions d'amélioration :

CanvasController:

- Permettre une sélection des éléments du canvas à l'aide du curseur. (carré de sélection). Cette étape passe par l'unification des interacteurs de CanvasController et Item, dans un seul et même interacteur au sein de CanvasControleur.
- Effectuer lors des sélections multiples d'Items sur le canvas le ré-attachage des Item à un GnomeCanvasGroup commun, de façon à déplacer plusieurs Item en même temps.
- Utiliser le widget sélection de couleur fournit par Gtk pour permettre à l'utilisateur d'affecter ses propres couleurs à l'environnement (et sauvegarder ses paramètres)
- Permettre à l'utilisateur de modifier le nom d'un Blend, en double cliquant dessus, par le biais d'un popup d'édition du nom.
- Implémenter un autoscroll sur le Canvas, de façon à ce qu'il se déroule naturellement lors du déplacement d'un objet.

Drag'N Drop:

- Permettre le Drag'N Drop de fichiers depuis n'importe quelle application (Nautilus ...)
- Permettre que lors d'un drag and drop, on puisse finalement ressortir le curseur du canvas sans pour autant y laisser un objet.

Editor:

- `_onSetanimCB()` est relativement basique pour le moment, il doit appeler un algorithme récursif pour rafraîchir tout l'arbre de blends jusqu'aux premiers parents.
- Cet algorithme choisit de façon arbitraire le nombre d'image-clef à donner à l'animation demandé lors du rafraîchissement d'un blend. Si les deux animations d'un blending ont une durée différente, l'utilisateur devrait pouvoir choisir quelle durée il souhaite avoir dans l'animation résultante.

Constraint Editor

Sur le Constraint Editor, seule la fonction Undo/ Redo reste à stabiliser, dans l'absolu, toutes les fonctionnalités du Constraint Editor prendront effet lors de l'évolution de la hLib.

Contacts :

Vincent Lachenal : winstu@gmail.com

Raphaël Ducom : rastakouer@gmail.com