Node Editor Roadmap

An underline of the current project status

by Seneral

Version 1.02 (14.02.16)

Forums

-

Repository

Roadmap Overview

TODO	Priority	State
Action Node (-> Editor Controls)	3	Active; Collab. of Kamigaku and Seneral
Nested Scaling support	1	On Hold; Need math. Help!
Save overhaul and extension (4)	2	Planned; Extension progressed
Undo support (Editor + Runtime)	2	Planned; Command-Pattern Undo integration in the works; RT: VFW
Additional Editor controls (6)	3	Planned
Bug Fixing/Checking	2	Few known Bugs
DONE RECENTLY		**State**
Statemachine Behaviour	1	'Beta'; **Please look out for bugs:)**

Statemachine Behaviour Support

- Description: Integration into the existing calculation system, enables for transitioning though nodes based on conditions like the mecanim statemachine
- State: Current Focus;
  Visually setup and transitions are integrated including conditions, what is missing is only the actual transitioning behaviour aswell as the GUI and methodics to create conditions and other related stuff.
- Difficulty/Scale: Moderate
- Location: NodeEditor.cs/Calculation and Transition.cs aswell as Node.cs

Nested Scaling Support

- Description: Enables support for Group Nodes
- State: On Hold because of mathematical problem; Open;
  Mathematical help needed!
- Difficulty/Scale: High
- Location: GUIScaleUtility.cs
- Current Problem: Nested Scale areas get shifted when atleast two groups in the hierarchy are scaled (the scale gets stacked), but the value is yet unknown.
- Problem State: Environment to test this out is already setup. Factor for this shift value is definitely the position of the nested group **relative to the topmost group**, that means each parent group aswell as it's own position influences the shift value. Scale values of each parent group obviously influence the value, too.
- Goal: Calculating the shift value with a formula. Eventually, we just need to transform the initial position in screen space according to ta any existing scales, accounting for pivots and other stuff... (tedious)

Save Overhaul and Extensions (4)

- **Per Node Saving** option to save additional stuff like Scriptableobjects along in the save file (Difficulty: Easy)
- **Save file inspector** with a simple button to open it and informations (Difficulty: Easy)
- **Better Auto save** similar to the save system before the overhaul *cbc73f3*: After saving, the canvas gets saved somewhere else, too, **without duplicating**, so that the link persists. Then most changes are automatically saved, only additional ScriptableObjects like Nodes need to be included manually. (Difficulty: Moderate)
- **Custom Binary or XML save format** for alternatice saving/loading at Runtime (Difficulty: High)
- State: **On Hold** because of different focus; **Open**

Undo Support

- Description: Undo support either based on the default system (tedious) or using my WIP Command-Pattern Undo Integration (yes, integration into the default system:D) when it's finished (Progressing good, but currently on hold). For Runtime, VFW would probably be the best choice as the undo system (also command-pattern)
- Difficulty/Scale: High
- State: **On Hold** because of different focus; **Open**;

Additional Editor Controls (6)

- **Multi-Selecting Nodes** (Difficulty: Moderate)
- **Grouping Nodes in Areas**. Meaybe just colored rect with a header that will serve as a parent to all attached nodes for organizing bigger canvases, just like in Substance Designer (Difficulty: Moderate-Easy)
- **Dynamic Node Rect Size**. No built-in way, manually returning size for each node is not an option; Using reflection it may be possible to get the current GUI Layouting stack. (Difficulty: High)
- **Finishing up OverlayGUI like Popups**, making use of the already implemented Popup Engine used by GenericMenu in OverlayGUI.cs (Difficulty: Moderate)
- **Improve Knob Customizations** like activation toggles, delegates to check for specific properties like color and texture and even drawing overrides (Difficulty: Moderate-Easy)
- **Generic 'Action Node'** which allows to call any function (both member and statics), input and outputs adjusting based on function parameters (Difficulty: High)
- State: On Hold because of different focus; **Open**

Bug Fixing/Checking

- Current Known Bug Reports. Also check the Repo issues.
- Check Bug regarding GL on Cameras mentioned in forum post #163
- Check Bug regarding Node Control Order mentioned in forum post #236

Note

You can ask me on any of these plans through a PM on the Unity forums (Seneral) or per Email (lev.gaeher@gmail.com).
We'd all also welcome contributions on tasks marked as Open or those who are explicitly marked with need help;)