

---

Subject: : Open Amiga

Topic: : Assigned: ReAction menu.class project

Re: Assigned: ReAction menu.class project

Author: : broadblues

Date: : 2014/7/1 12:43:42

URL:

@trixie

Looking at that I can't see the menu structure as easily as looking at the equivalent Gadtools code.

Though that might be solved as simple as creating aliases for MENU\_AddChild as in MENU\_AddMenu  
MENU\_AddSubMenu MENU\_AddItem etc etc

Is an object deciding it's nature by it position in the heriachy not going to be bit of a headache coding wise?

What if someone detaches a menu object and tries to add it at a different point in the heirarchy ? Or has sets of menus ready to swap in and out depending on context (add a paint tool menu when the paint tool is active remove it and add a erase tools menu etc etc) with your concept as I read it while detached the top of the heirarchy would be a strip then when attached to the strip it would suddenly become a Menu. Possible I'm sure but a lot interobject communication required.

What about specifying the type (default might be MENU\_ITEM or what ever turns out to be most sensible)

```
menu[STRIP} = MenuObject.  
    MENU_Type,MENU_STRIP,  
    MENU_AddChild, Menu[PROJECT]= MenuObject,  
        MENU_Type, MENU_MENU,  
        MENU_Label, "Project",  
    MENU_AddChild, MenuObject,  
        MENU_Type,MENU_ITEM,  
        MENU_Label, "foo",  
    END,  
END,  
END;
```

```
Menu[PAINT] = MenuObject.  
    MENU_Type,MENU_MENU,  
    ....  
END;
```

```
if newtool = PAINT
```

DoMethod(Menu[STRIP], MM\_INSERT\_ITEM\_AFTER, Menu[PAINT],MENU[PROJECT])

Random feature thought, a menu object of type strip (however that is determined) could take an object of Window as an attribute then could handle attachment and removal transparently when methods modifying the menu are invoked.