
Subject: Re: Proposed change to U++ to allow owning children.

Posted by [Lance](#) on Wed, 16 Mar 2011 16:39:22 GMT

[View Forum Message](#) <> [Reply to Message](#)

kohait00:

I have implemented the requested changes. Just replace existing files with my attached version (make backup before doing that) and you should be able to use it immediately.

Here is some sample how you can use it.

```
struct MyButton : Moveable<MyButton>, Button
{
    // note: always default to false because
    // existing U++ users are used to this behaviour
    //
    MyButton(bool toBeOwned=false)
    {
        SetOwned(toBeOwned);
    }

    virtual ~MyButton()
    {
        DUMP("Yes, I am properly destructed!!!");
    }
};

class App : public TopWindow
{
public:
    App()
    {
        Button * p=new MyButton(true);
        Button * p2=new MyButton();

        p->SetLabel("Button 1").TopSizeZ(2,20).LeftSize(2, 60);
        p2->SetLabel("Button 2").TopSizeZ(40,20).LeftSize(2, 60);
        Add(*p); // p will be destructed because we SetOwned in
                // its constructor;
        // or (*this) << p;

        Add(*p2, 1);
    }
};
```

Please try it and let me know there is any problem. I am not very sure if I changed anything in

Ctrl.cpp.
