
Subject: Re: More GLCtrl

Posted by [mdelfede](#) on Mon, 20 Oct 2008 12:26:49 GMT

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

luzr wrote on Sun, 19 October 2008 18:20l believe all should be fix now.

Note: DHCtrl was never intended to have children... Therefore, rather than rewriting half of CtrlCore, I have solved the issue by restructuring GLCtrl - it has now DHCtrl as child (and derives from ParentCtrl).

Thanks for hint and a quality testcase - I have put it to upptst for further use

Mirek

Eh, do you remember when some monthes ago I told you CtrlCore would need some strong refactoring ?

Writing Linux GLCtrl was a nightmare because of the impossibility of having childs of windowed controls. I introduced them with some dirty hacks on Ctrl stuffs trying to not break previous behaviours, but just to Linux part.... and it was quite complicated because of optimizations in ctrlcore classes. And I know it's not a perfect implementation.....

I've seen some similar (but differently handled) hacks on windows DHCtrl code.

But I agree with you, it would be a *big* effort to clean up CtrlCore classes.

For other people not knowing the problem, OpenGL needs an underlying OS window (X11 or Windows). The problem arose because UPP controls don't have an underlying window (besides top ones) for speed (and other) purposes and were not foreseen to contain controls with a window handler.

All CtrlCore classes assume that no child control can have a window handle, so they use often the fact that there's no parent window to assume the control is a top one; in case of DHCtrl that's wrong. All hacks on linux part were aimed to remove this assumption.

Max

EDIT : btw, IMHO, the best would be to go in the direction on which child windowed controls are allowed. That's done on Linux part, it was not easy and it's not completely well implemented, but can be done. That would be a more uniform behaviour and would allow to embed some fancy windowed controls without the need of a wrapper class, and allowing them to be also containers for upp controls. (Linux DHCtrl can do almost all about it).

Max
