Subject: Re: [Proposal] Adding a GLLock struct to GLCtrl
Posted by Klugier on Wed, 02 Sep 2020 22:37:34 GMT
View Forum Message <> Reply to Message

Hello Xemuth,

From my point of view the idea of "locking" looks reasonable. However, we should make it more smarter. Please noticed that the API we commit will be with us for the very long time. So, we need to be careful here. All we need to do is make context current and release it at the end of the block. So, I would suggest adding three dedicated methods to context attaching/activating GLCtrl and buffer swapping:
- ActivateContext(); (Anyway void ActivateContext(); from Win32 GLPane seems unimplemented)
- DeactivateContext();
- SwapBuffers();

So, in your case the logic will look as follow:

```
canvas.ActivateContext();

//Shaders Stuff
//OpenGL buffer filling and loading

canvas.SwapBuffers();
canvas.DeactivateContext();
// All these method will be proxy and send work to GLPane's classes.

return true;
```

In case of simplification we should add lock mechanisms as you suggested (The three above method should still be available, so you will have a choice):

```
GLCtrl::ContextLock __(canvas); // <- If this is nested glass GL prefix is redundant.

// Do we always want buffer swpaing if not I would see additional lock...
GLCtrl::ContextLockWithSwapBuffers __(canvas); // <- Alternatively ContextLock can have additional bool parameter (true by default),
                                               // however according to the clean code bool parameter should be avoided, so I
                                               // use more verbose notation here.
```

We also need documentation for GLCtrl and it seems that Copying file is missing too. Mirek can we add the second file?

Klugier