
Subject: Re: GLCtrl in GTK

Posted by [mirek](#) on Mon, 03 Dec 2018 07:54:15 GMT

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

ptkacz wrote on Thu, 29 November 2018 13:33Mirek,

Upon making the code changes and building, I first tested the demo without the NOGTK and was presented with a blank demo window. When specifying NOGTK + GUI application, the demo ran.

:(

Well, we better have this fixed. I hope you will help me, as your setup seems to be the only one failing.

Can you test this? This is boilerplate GLX code from the internet that was my starting point in developing new GLCtrl incarnation:

```
#include <Core/Core.h>
```

```
#include <GL/glx.h>
```

```
#include <GL/gl.h>
```

```
#include <unistd.h>
```

```
#include <iostream>
```

```
#include <stdlib.h>
```

```
#include <string.h>
```

```
#include <stdio.h>
```

```
#define GLX_CONTEXT_MAJOR_VERSION_ARB    0x2091
```

```
#define GLX_CONTEXT_MINOR_VERSION_ARB    0x2092
```

```
typedef GLXContext (*glXCreateContextAttribsARBProc)(Display*, GLXFBConfig, GLXContext, Bool, const int*);
```

```
CONSOLE_APP_MAIN
```

```
{
    Display *display = XOpenDisplay(0);

    glXCreateContextAttribsARBProc glXCreateContextAttribsARB = NULL;

    const char *extensions = glXQueryExtensionsString(display, DefaultScreen(display));
    std::cout << extensions << std::endl;

    static int visual_attribs[] =
    {
        GLX_RENDER_TYPE, GLX_RGBA_BIT,
        GLX_DRAWABLE_TYPE, GLX_WINDOW_BIT,
```

```

    GLX_DOUBLEBUFFER, true,
    GLX_RED_SIZE, 1,
    GLX_GREEN_SIZE, 1,
    GLX_BLUE_SIZE, 1,
    None
};

std::cout << "Getting framebuffer config" << std::endl;
int fbcount;
GLXFBConfig *fbc = glXChooseFBConfig(display, DefaultScreen(display), visual_attribs,
&fbcount);
if (!fbc)
{
    std::cout << "Failed to retrieve a framebuffer config" << std::endl;
    return;
}

std::cout << "Getting XVisualInfo" << std::endl;
XVisualInfo *vi = glXGetVisualFromFBConfig(display, fbc[0]);

XSetWindowAttributes swa;
memset(&swa, 0, sizeof(swa));
std::cout << "Creating colormap" << std::endl;
swa.colormap = XCreateColormap(display, RootWindow(display, vi->screen), vi->visual,
AllocNone);
swa.border_pixel = 0;
swa.event_mask = 0;//StructureNotifyMask;

std::cout << "Creating window" << std::endl;
Window win = XCreateWindow(display, RootWindow(display, vi->screen), 0, 0, 100, 100, 0,
vi->depth, InputOutput, vi->visual, CWBorderPixel|CWColormap|CWEventMask, &swa);
if (!win)
{
    std::cout << "Failed to create window." << std::endl;
    return;
}

std::cout << "Mapping window" << std::endl;
XMapWindow(display, win);

// Create an oldstyle context first, to get the correct function pointer for
glXCreateContextAttribsARB
GLXContext ctx_old = glXCreateContext(display, vi, 0, GL_TRUE);
glXCreateContextAttribsARB = (glXCreateContextAttribsARBProc)glXGetProcAddress((const
GLubyte*)"glXCreateContextAttribsARB");
glXMakeCurrent(display, 0, 0);
glXDestroyContext(display, ctx_old);

```

```

if (glXCreateContextAttribsARB == NULL)
{
    std::cout << "glXCreateContextAttribsARB entry point not found. Aborting." << std::endl;
    return;
}

static int context_attribs[] =
{
    GLX_CONTEXT_MAJOR_VERSION_ARB, 3,
    GLX_CONTEXT_MINOR_VERSION_ARB, 0,
    None
};

/*
std::cout << "Creating context" << std::endl;
GLXContext ctx = glXCreateContextAttribsARB(display, fbc[0], NULL, true, context_attribs);
if (!ctx)
{
    std::cout << "Failed to create GL3 context." << std::endl;
    return;
}
*/
// GLXContext ctx = glXCreateContextAttribsARB(display, fbc[0], NULL, true, context_attribs);
GLXContext ctx = glXCreateContext(display, vi, NULL, GL_TRUE);

std::cout << "Making context current" << std::endl;
glXMakeCurrent(display, win, ctx);

for(int i = 0; i < 4; i++) {
    glClearColor (0, 0.5, 1, 1);
    glClear (GL_COLOR_BUFFER_BIT);
    glXSwapBuffers (display, win);

    sleep(1);

    glClearColor (1, 0.5, 0, 1);
    glClear (GL_COLOR_BUFFER_BIT);
    glXSwapBuffers (display, win);

    sleep(1);
}
ctx = glXGetCurrentContext();
glXMakeCurrent(display, 0, 0);
glXDestroyContext(display, ctx);
}

```

It should create a window and blink it 4 times...

Also, can you please .zip your GLCtrl package with changes applied and post here?

Thanks,

Mirek
