Subject: GLDraw
Posted by mirek on Mon, 16 Sep 2013 08:54:29 GMT
View Forum Message <> Reply to Message

New package GLDraw implements SDraw based Draw in OpenGL, as demonstrated in reference/GLDrawDemo.

Subject: Re: GLDraw
Posted by koldo on Mon, 16 Sep 2013 17:35:06 GMT
View Forum Message <> Reply to Message

Hello Mirek

Compiling with MSC10 in Win7 I get these errors:

'GL_BGRA' : undeclared identifier
'GL_CLAMP_TO_EDGE' : undeclared identifier

Subject: Re: GLDraw
Posted by Novo on Tue, 17 Sep 2013 04:32:28 GMT
View Forum Message <> Reply to Message

koldo wrote on Mon, 16 September 2013 13:35Hello Mirek

Compiling with MSC10 in Win7 I get these errors:

'GL_BGRA' : undeclared identifier
'GL_CLAMP_TO_EDGE' : undeclared identifier

Same with me.

Subject: Re: GLDraw
Posted by Novo on Tue, 17 Sep 2013 13:11:41 GMT
View Forum Message <> Reply to Message

On Linux:

Subject: Re: GLDraw
Posted by Novo on Tue, 17 Sep 2013 17:02:41 GMT

View Forum Message <> Reply to Message

After fixing compilation it is flickering like crasy when resizing window in both Windows and Linux. And also size of centered text is changing ...

Subject: Re: GLDraw
Posted by mirek on Thu, 19 Sep 2013 18:09:43 GMT
View Forum Message <> Reply to Message

Novo wrote on Tue, 17 September 2013 13:02After fixing compilation it is flickering like crasy when resizing window in both Windows and Linux. And also size of centered text is changing ...

Centering should be now fixed. Flickering is of not consequence, it is just quick example using GLCtrl, real usage will be different...

Subject: Re: GLDraw
Posted by Novo on Sat, 21 Sep 2013 02:55:51 GMT
View Forum Message <> Reply to Message

Flickering is also gone ...
Thanks!

Subject: Re: GLDraw
Posted by Tom1 on Sun, 01 Dec 2013 20:47:53 GMT
View Forum Message <> Reply to Message

Hi,

I just tried the current reference/GLDrawDemo for the first time and run into trouble on Windows Vista Business 64 bit, using MSC9 32 bit:

1. In main.cpp I had to switch the order of includes:
#include <GLDraw/GLDraw.h>
#include <GLCtrl/GLCtrl.h>

because of this error:

main.cpp
c:\upp\uppsrc\gldraw\glew.h(84) : fatal error C1189: #error : gl.h included before glew.h
GLDrawDemo: 1 file(s) built in (0:02.34), 2343 msecs / file, duration = 2375 msecs

2. In GLDrawS.cpp I had to rename two variables (called near and far) to something else, since

confused the MSC9 32-bit compiler:

void GLOrtho(float left, float right, float bottom, float top, float nnear, float nfar, GLuint u_projection)

3. Finally, when linking I got this:

```
Linking...
GLDraw.lib(GLDrawS.obj) : error LNK2019: unresolved external symbol __imp____glewGetUniformLocation referenced in function "public: int __thiscall Upp::GLProgram::GetUniform(char const *)" (?GetUniform@GLProgram@Upp@@QAEHPBD@Z)
GLDraw.lib(GLDrawS.obj) : error LNK2019: unresolved external symbol __imp____glewUseProgram referenced in function "public: void __thiscall Upp::GLProgram::Use(void)" (?Use@GLProgram@Upp@@QAEXXZ)
GLDraw.lib(GLShaders.obj) : error LNK2001: unresolved external symbol __imp____glewUseProgram
GLDraw.lib(GLDrawS.obj) : error LNK2019: unresolved external symbol __imp____glewEnableVertexAttribArray referenced in function "void __cdecl Upp::initializeGL(void)" (?initializeGL@Upp@@YAXXZ)
GLDraw.lib(GLDrawS.obj) : error LNK2019: unresolved external symbol __imp____glewUniform1i referenced in function "void __cdecl Upp::initializeGL(void)" (?initializeGL@Upp@@YAXXZ)
GLDraw.lib(GLDrawS.obj) : error LNK2019: unresolved external symbol __imp__glewInit referenced in function "void __cdecl Upp::initializeGL(void)" (?initializeGL@Upp@@YAXXZ)
GLDraw.lib(GLDrawS.obj) : error LNK2019: unresolved external symbol __imp____glewUniformMatrix4fv referenced in function "void __cdecl Upp::GLOrtho(float,float,float,float,float,float,unsigned int)" (?GLOrtho@Upp@@YAXMMMMMMI@Z)
GLDraw.lib(GLDrawS.obj) : error LNK2019: unresolved external symbol __imp____glewDisableVertexAttribArray referenced in function "private: void __thiscall Upp::GLDraw::FlushPutRect(void)" (?FlushPutRect@GLDraw@Upp@@AAEXXZ)
GLDraw.lib(GLDrawS.obj) : error LNK2019: unresolved external symbol __imp____glewVertexAttribPointer referenced in function "private: void __thiscall Upp::GLDraw::FlushPutRect(void)" (?FlushPutRect@GLDraw@Upp@@AAEXXZ)
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol __imp____glewDeleteShader referenced in function "public: void __thiscall Upp::GLProgram::Clear(void)" (?Clear@GLProgram@Upp@@QAEXXZ)
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol __imp____glewDeleteProgram referenced in function "public: void __thiscall Upp::GLProgram::Clear(void)" (?Clear@GLProgram@Upp@@QAEXXZ)
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol __imp____glewGetShaderInfoLog referenced in function "unsigned int __cdecl Upp::LoadShader(char const *,unsigned int)" (?LoadShader@Upp@@YAIPBDI@Z)
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol
```

```
__imp____glewGetShaderiv referenced in function "unsigned int __cdecl Upp::LoadShader(char
const *,unsigned int)" (?LoadShader@Upp@@YAIPBDI@Z)

GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol
__imp____glewCompileShader referenced in function "unsigned int __cdecl
Upp::LoadShader(char const *,unsigned int)" (?LoadShader@Upp@@YAIPBDI@
 Z)
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol
__imp____glewShaderSource referenced in function "unsigned int __cdecl Upp::LoadShader(char
const *,unsigned int)" (?LoadShader@Upp@@YAIPBDI@Z
 )
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol
__imp____glewCreateShader referenced in function "unsigned int __cdecl Upp::LoadShader(char
const *,unsigned int)" (?LoadShader@Upp@@YAIPBDI@Z
 )
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol
__imp____glewGetProgramInfoLog referenced in function "public: void __thiscall
Upp::GLProgram::Create(char const *,char const *,struct Upp::Tu
 ple2<int,char const *> *,int)"
(?Create@GLProgram@Upp@@QAEXPBD0PAU?$Tuple2@HPBD@2@H@Z)
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol
__imp____glewGetProgramiv referenced in function "public: void __thiscall
Upp::GLProgram::Create(char const *,char const *,struct Upp::Tuple2<
 int,char const *> *,int)"
(?Create@GLProgram@Upp@@QAEXPBD0PAU?$Tuple2@HPBD@2@H@Z)
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol
__imp____glewLinkProgram referenced in function "public: void __thiscall
Upp::GLProgram::Create(char const *,char const *,struct Upp::Tuple2<i
 nt,char const *> *,int)"
(?Create@GLProgram@Upp@@QAEXPBD0PAU?$Tuple2@HPBD@2@H@Z)
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol
__imp____glewBindAttribLocation referenced in function "public: void __thiscall
Upp::GLProgram::Create(char const *,char const *,struct Upp::T
 uple2<int,char const *> *,int)"
(?Create@GLProgram@Upp@@QAEXPBD0PAU?$Tuple2@HPBD@2@H@Z)
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol
__imp____glewAttachShader referenced in function "public: void __thiscall
Upp::GLProgram::Create(char const *,char const *,struct Upp::Tuple2<
 int,char const *> *,int)"
(?Create@GLProgram@Upp@@QAEXPBD0PAU?$Tuple2@HPBD@2@H@Z)
GLDraw.lib(GLShaders.obj) : error LNK2019: unresolved external symbol
__imp____glewCreateProgram referenced in function "public: void __thiscall
Upp::GLProgram::Create(char const *,char const *,struct Upp::Tuple2
 <int,char const *> *,int)"
(?Create@GLProgram@Upp@@QAEXPBD0PAU?$Tuple2@HPBD@2@H@Z)
C:\upp\out\reference\MSC9.Gui.Sse2\GLDrawDemo.exe : fatal error LNK1120: 21 unresolved
externals
```

There were errors. (0:04.78)

Can someone help me with this?

Best regards,

Tom

Subject: Re: GLDraw
Posted by mirek on Mon, 02 Dec 2013 16:41:24 GMT
View Forum Message <> Reply to Message

Should be now fixed.

Mirek

Subject: Re: GLDraw
Posted by Tom1 on Tue, 03 Dec 2013 14:50:48 GMT
View Forum Message <> Reply to Message

Thanks Mirek. 1. and 2. are now fixed, but the linking problem 3. is still there.

Tom

Subject: Re: GLDraw
Posted by Tom1 on Tue, 03 Dec 2013 22:08:16 GMT
View Forum Message <> Reply to Message

OK, I'm now one step further after adding GLDraw package compiler option: WIN32: /DGLEW_STATIC

Now the demo compiles/links/runs, but with an empty (black) window interior.

How to proceed?

Best regards,

Tom

Subject: Re: GLDraw
Posted by mirek on Wed, 04 Dec 2013 13:34:03 GMT

View Forum Message <> Reply to Message

Tom1 wrote on Tue, 03 December 2013 17:08OK, I'm now one step further after adding GLDraw package compiler option: WIN32: /DGLEW_STATIC

Now the demo compiles/links/runs, but with an empty (black) window interior.

How to proceed?

Best regards,

Tom

It now works for me (with some linker warnings) just as it is in repositry. MSC9/Win7.

Anything specific about your OS?

Subject: Re: GLDraw
Posted by Tom1 on Wed, 04 Dec 2013 14:33:35 GMT
View Forum Message <> Reply to Message

Mirek,

I have now tested it on both Windows Vista Business 64 bit and Windows 7 Professional 64-bit. Both absolutely must have the GLEW_STATIC defined. Otherwise, I will get a large amount of compiler warnings for glew related stuff and also the 21 linker errors I reported previously. I'm compiling with MSC9 32-bit.

After defining the GLEW_STATIC, there are no such warnings or errors at all during GLDraw compilation nor executable linking. (Through GLDraw package compiler option WIN32: /DGLEW_STATIC).

--

Maybe the screen update problems are related to the low-end NVidia GeForce 6150 display adapter. Or maybe I should have some extra GL drivers for it on windows?

Best regards,

Tom

Subject: Re: GLDraw
Posted by Tom1 on Wed, 04 Dec 2013 15:04:49 GMT
View Forum Message <> Reply to Message

OK, solved -- sort of. It works after I comment the following line in GLDraw.h:

#define GL_USE_SHADERS

So, I guess the old GeForce 6150 is too old and does not support shaders.

Best regards,

Tom

Subject: Re: GLDraw
Posted by ManfredHerr on Fri, 13 Mar 2015 16:55:24 GMT
View Forum Message <> Reply to Message

My computer runs both, WIN / XP prof. and UBUNTU 14.04 alternatively. The graphics card is a NVIDIA GT9600.
GLDraw demo compiles and runs on WIN / XP if TheIDE is adviced to use MSVC10.0 for building. MSVC8, that is installed as well, and used by default, gives an error about SPI_GETWHEELSCROLLINES.
Using UBUNTU I could not manage to see the same result. Here, I use gcc as build environment. Until I updated the driver to the latest propietary (NVIDIA) driver. I got error messages "shaders do not compile" or "heap leaks detected". Now, build and run gives no error messages but the window shows pure black, like Tom's experience. For a short moment the rectangle and two lines in the upper left corner appear before all is painted black.
Am I missing something?
