Subject: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Klugier on Sat, 29 Aug 2020 16:41:08 GMT
View Forum Message <> Reply to Message

Hello,

It seems that there is new glew version (2.2.0 - March 2020). We should updated it before releasing new Upp version. Also, I had discussion today with Xemuth and our current version forces him to use GLAD.

In attachment I attached plugin/glew updated to version 2.2.0. If you want to dowload it by yourself you could do this on Sourceforge.

If you are interested in this plugin. Please test it Especially I would be grateful if Xemuth test this plugin by himself.

Sincerely,
Klugier

File Attachments
1) glew.zip, downloaded 365 times

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Xemuth on Sat, 29 Aug 2020 17:25:02 GMT
View Forum Message <> Reply to Message

I will test it now. Will be back soon

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Xemuth on Sat, 29 Aug 2020 18:00:36 GMT
View Forum Message <> Reply to Message

First I have changed the "uppsrc/plugin/glew" by the glew you provided, then I have launch the example : Reference->OpenGL
Compilation error: C:\Upp\upp\uppsrc\plugin\glew\glew.c (34): fatal error: 'GL/glew.h' file not found

Comparing the glew you provided with the one Upp had, it should be #include "glew.h"
Same for line 53 in glew.c it should be #include "wglew.h" instead of #include <GL/wglew.h>
Same for....
You will find the patch file bellow.

Also, GLCtrl have a strange behavior concerning context (A behavior I changed in my own GLCtrl) :

When ExecuteGL is call, in original version it do this :

```
void GLCtrl::GLPane::ExecuteGL(HDC hDC, Event<> paint, bool swap_buffers){
    /**** CODE ****/
    wglMakeCurrent(hDC, s_openGLContext); //Hooking the OpenGL Context
   /***** Code / Swap buffer etc .... *****/
   wglMakeCurrent(NULL, NULL); //Releasing the current OpenGL Context
}
```

This aquisition then releasing is problematique since it prevent you to execute some OpenGL code outside of GLPaint function.
It mean I can't do OpenGL when button is pressed or fenetre resized or even when my application is starting etc... To fix this issue, in my own Ctrl I just changed the fonction like this :

```
void GLCtrl::GLPane::ExecuteGL(HDC hDC, Event<> paint, bool swap_buffers)
{
 ONCELOCK{
     wglMakeCurrent(hDC, s_openGLContext); //Creating the context one time
 }
 /**** CODE .... ****/
 //wglMakeCurrent(NULL, NULL); //no release
}
```

Since I encoutered no problem with my version and I don't see (maybe I'm wrong) any problem with it, I would be happy it become so official behavior of GLCtrl. So I could migrate my surfaceCtrl package to this GLCtrl.

With all this, my application and OpenGL Example (in reference) work perfectly.

File Attachments

1) glew.c.patch, downloaded 391 times
2) Win32GLCtrl.cpp.patch, downloaded 392 times
3) XGLCtrl.cpp.patch, downloaded 385 times

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Xemuth on Sat, 29 Aug 2020 18:22:38 GMT
View Forum Message <> Reply to Message

I have noticed a big chunk of warning using it :

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Klugier on Sat, 29 Aug 2020 19:06:54 GMT
View Forum Message <> Reply to Message

Hello Xemuth,

Thanks for the feedback. I compiled glew on windows and apply patches + fix the issue with compilation warning. Right now glew is always compile in static mode. I just wonder why "New add remove/flag" options doesn't work (with single GLEW_STATIC option) and I need manually add compiler option -D GLEW_STATIC. Mirek if you read that thread - could you answer to that question.

Xemuth small power tip for you. Keeping commented code on production is treated as code smell. Please avoid this behavior in your future commits. You can read more about this for example here or just search for "commented code" in Google

I would suggest Mirek to test and apply your fixes to OpenGL you posted in this thread (of course without commented code). Thanks for these improvements!

Sincerely,
Klugier

File Attachments

1) glew.zip, downloaded 370 times

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by mirek on Sun, 30 Aug 2020 17:42:39 GMT
View Forum Message <> Reply to Message

Xemuth wrote on Sat, 29 August 2020 20:00
Also, GLCtrl have a strange behavior concerning context (A behavior I changed in my own GLCtrl) :

When ExecuteGL is call, in original version it do this :

```
void GLCtrl::GLPane::ExecuteGL(HDC hDC, Event<> paint, bool swap_buffers){
    /**** CODE ****/
    wglMakeCurrent(hDC, s_openGLContext); //Hooking the OpenGL Context
   /***** Code / Swap buffer etc .... *****/
   wglMakeCurrent(NULL, NULL); //Releasing the current OpenGL Context
}
```

This aquisition then releasing is problematique since it prevent you to execute some OpenGL code outside of GLPaint function.

This is needed so that you can have multiple GLCtrls in multiple windows. If you need to execute OpenGL code outside of GLPaint function, just use ExecuteGL (it is in fact inteded for that use).

Mirek

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Klugier on Sun, 30 Aug 2020 20:04:36 GMT
View Forum Message <> Reply to Message

Hello Mirek,

It seems that we have all information to update glew. Howeverm what about this change proposed by Xemuth:

if (glewIsSupported("GL_VERSION_4_3")) enhanced_mode=true;

instead of

if (glewIsSupported("GL_VERSION_2_1")) enhanced_mode=true;

Do we need it? BTW, This string should be defined in one place to change it more easily. Right now to change it we need to modify two source files (Win & X11). Mire, can we improve this? I do not have access to GLCtrl.

Klugier

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by mirek on Sun, 30 Aug 2020 22:01:25 GMT
View Forum Message <> Reply to Message

Klugier wrote on Sun, 30 August 2020 22:04Hello Mirek,

It seems that we have all information to update glew.

Updated.

Quote:

if (glewIsSupported("GL_VERSION_4_3")) enhanced_mode=true;

instead of

if (glewIsSupported("GL_VERSION_2_1")) enhanced_mode=true;

Do we need it?

No. Check the code for the real purpose. Above change would be grave mistake.

Quote:
BTW, This string should be defined in one place to change it more easily. Right now to change it we need to modify two source files (Win & X11).

I do not see it in X11. This is win32 specific issue if I remember well...

Mirek

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Klugier on Sun, 30 Aug 2020 22:23:39 GMT
View Forum Message <> Reply to Message

Hello Mirek,

Thanks for updating this plugin is such short amount of time!

You are right about GL version string it is only used in Windows implementation. About the if you are right too, we will do not execute some enhanced OpenGL initialization when version will be less than 4.3. If this initialization works correctly on 2.1 there is no need to change them.

Klugier

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Xemuth on Sun, 30 Aug 2020 22:35:10 GMT
View Forum Message <> Reply to Message

mirek wrote on Sun, 30 August 2020 19:42Xemuth wrote on Sat, 29 August 2020 20:00
Also, GLCtrl have a strange behavior concerning context (A behavior I changed in my own GLCtrl) :

When ExecuteGL is call, in original version it do this :

```
void GLCtrl::GLPane::ExecuteGL(HDC hDC, Event<> paint, bool swap_buffers){
   /**** CODE ****/
   wglMakeCurrent(hDC, s_openGLContext); //Hooking the OpenGL Context
```

```
  /***** Code / Swap buffer etc .... *****/
  wglMakeCurrent(NULL, NULL); //Releasing the current OpenGL Context
}
```

This aquisition then releasing is problematique since it prevent you to execute some OpenGL code outside of GLPaint function.

This is needed so that you can have multiple GLCtrls in multiple windows. If you need to execute OpenGL code outside of GLPaint function, just use ExecuteGL (it is in fact inteded for that use).

Mirek

Indeed it work, thanks for the tips

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Xemuth on Sun, 30 Aug 2020 23:58:07 GMT
View Forum Message <> Reply to Message

I moved my package SurfaceCtrl to GLCtrl with the fresh Glew implementation. Everythings worked fine (CLANG / MSVS) in debug mode,
however when trying the Release mode linking error occure:

The MSVS error:

The CLANG one :

Apparently this problem occure when compilation options GLEW_STATIC is not defined but in the case of plugin/glew it is actually defined so I don't know why it occure

---

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Xemuth on Mon, 31 Aug 2020 00:28:23 GMT
View Forum Message <> Reply to Message

The linking error is fix by adding this define at the top of GLCtrl.h
#define GLEW_STATIC
Just before the glew include.

However I don't understand why it work since it is still defined as compilation option in package configuration of plugin/glew....

Update: After trying to set -D GLEW_STATIC as compiler option in GLCtrl package configuration it worked. Maybe this compiler option should be move from plugin/glew to GLCtrl

---

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Klugier on Mon, 31 Aug 2020 20:30:18 GMT
View Forum Message <> Reply to Message

Hello Xemuth,

It seems that Mirek push fix with GLEW_STATIC. Also OepnGL_gald can be removed from bazzar to do not produced unnecessary confusion which GLCtrl should be used. Will you remove it?

plugin/glad also seems unnecessary now. And trust me the less package to maintain the better

Klugier

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Xemuth on Tue, 01 Sep 2020 00:58:29 GMT
View Forum Message <> Reply to Message

Klugier wrote on Mon, 31 August 2020 22:30Hello Xemuth,

It seems that Mirek push fix with GLEW_STATIC. Also OepnGL_gald can be removed from bazzar to do not produced unnecessary confusion which GLCtrl should be used. Will you remove it?

plugin/glad also seems unnecessary now. And trust me the less package to maintain the better

Klugier

I agree, I will delete all glad package from bazaar

Update: all glad content has been removed

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Novo on Tue, 01 Sep 2020 23:03:41 GMT
View Forum Message <> Reply to Message

Compilation of reference/GLDrawDemo is broken on Linux.
In file included from
/home/ssg/.local/soft/bb-worker/worker/l-upp/build/uppsrc/plugin/glew/glew.c:55:
/home/ssg/.local/soft/bb-worker/worker/l-upp/build/uppsrc/plugin/glew/glxew.h:103:12: fatal error: 'GL/glew.h' file not found
include <GL/glew.h>
^~~~~~~~~~
It was broken in
plugin/glew: updated to 2.2.0 - March 2020
git-svn-id: svn://ultimatepp.org/upp/trunk@14939 f0d560ea-af0d-0410-9eb7-867de7ffcac7

Before that everything was fine.

Subject: Re: Updating GLEW to version 2.2.0 for 2020.2 Release
Posted by Klugier on Tue, 01 Sep 2020 23:14:56 GMT
View Forum Message <> Reply to Message

Hello Novo,

Thanks for reporting. I checked this example before creating this message, however I need to have glew library installed on my system. In this case all we need to do is replace in the plugin/glew source code:

include <GL/glew.h>

with

include "glew.h"

Let's do the same for eglew.h (line 110). Mirek please apply I do not have write permission for plugins...

In case of glew static I just wonder what is the difference between passing argument like "-D GLEW_STATIC" to "-DGLEW_STATIC". I see that plugin/png uses the second notation.

Klugier
