Subject: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by ptkacz on Mon, 26 Jun 2017 03:03:27 GMT
View Forum Message <> Reply to Message

Hi Guys,

Upon bringing up TheIDE, I launch the OpenGL demo from the reference section.

Upon compiling the running the program, the program is launched followed by window popping up with the following message:

Fatal error
Heap leaks detected!

Upon selecting the, "okay" button, all windows are closed.

The issue appears to be happening on the win.Open():

```
GUI_APP_MAIN
{
 Ctrl::GlobalBackPaint();
 TopWindow win;
 OpenGLExample gl;
 gl.SetFrame(InsetFrame());
 win.Add(gl.HSizePos(10, 10).VSizePos(10, 10));
 win.Sizeable().Zoomable();
    win.Open();
 // win.Run();
}
```

Anyone have some code that works demoing a simple OpenGL program?

Peter

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by Klugier on Mon, 26 Jun 2017 14:57:19 GMT
View Forum Message <> Reply to Message

Hello,

Please check following fix: http://ultimatepp.org/redmine/issues/1481 and let me know it helps to you. Please modify your internal upp code according to the diff posted on the page (attachment).

Sincerely,
Klugier

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by ptkacz on Tue, 27 Jun 2017 01:05:56 GMT
View Forum Message <> Reply to Message

Hi Klugier,

Thanks for the reply. Just curious, the line:

+ if(backpaint == FULLBACKPAINT || globalbackpaint && !hasdhctrl && !dynamic_cast<DHCtrl *>(this)*/) {

the "*/" before the closing parentheses appears to be at the end of a comment block?

So when I've completed the changes, then I just, "make" followed by a "make install"?

I'm guessing that this fix is currently in the current nightly build?

When's 2017.2 targeted for by the way?

Peter

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by Klugier on Tue, 27 Jun 2017 10:48:04 GMT
View Forum Message <> Reply to Message

Hello,

Please remove '/*' from the if statement and let me know it works. This patch is not included in nightly builds, so you need to apply it manually. I will need to talk with Mirek why this line is commented. It causes bad regression in OpenGL.

Please be careful with OpenGL it can works unstable in latest upp stable release.

Sincerely,
Klugier

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by ptkacz on Thu, 29 Jun 2017 03:13:04 GMT
View Forum Message <> Reply to Message

Hi Klugier,

Just thinking that each copy of U++ should come with a AMD Threadripper!

I did a complete fresh install. Also before running make, I made the changes as recommended (below). The changes had no impact and the same error was encountered.

Peter

```
void Ctrl::UpdateArea0(SystemDraw& draw, const Rect& clip, int backpaint)
{
 GuiLock __;
 LTIMING("UpdateArea");
 LLOG("========== UPDATE AREA " << UPP::Name(this) << " ==========");
 // ExcludeDHCtrls(draw, GetRect().GetSize(), clip);
 if(globalbackbuffer) {
  CtrlPaint(draw, clip);
  LLOG("========== END (TARGET IS BACKBUFFER)");
  return;
 }
          // if(backpaint == FULLBACKPAINT || globalbackpaint/* && !hasdhctrl &&
!dynamic_cast<DHCtrl *>(this)*/) {
    if(backpaint == FULLBACKPAINT || globalbackpaint && !hasdhctrl && !dynamic_cast<DHCtrl
*>(this)) {
  ShowRepaintRect(draw, clip, LtRed());
  BackDraw bw;
  bw.Create(draw, clip.GetSize());
  bw.Offset(-clip.TopLeft());
  bw.SetPaintingDraw(draw, clip.TopLeft());
  CtrlPaint(bw, clip);
  bw.Put(draw, clip.TopLeft());
  LLOG("========== END (FULLBACKPAINT)");
  return;
 }
```

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by Klugier on Thu, 29 Jun 2017 10:07:44 GMT
View Forum Message <> Reply to Message

Hello,

OK - I will check this problem. Please give me an information about your operating system setup, maybe this is the bug related to the Windows implementation. I was tested OpenGL with patch on Linux and it worked fine.

Also, please check compiling the OpenGL example using release mode instead of debug. It should silence "Heap leaks" error detection.

Can you explain to me - what do you have on mind when writing that "U++ should come with a AMD Threadrippe"?

Sincerely,
Klugier

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by ptkacz on Fri, 30 Jun 2017 01:28:24 GMT
View Forum Message <> Reply to Message

Regarding the AMD Threadripper, just some humor. I was thinking how much faster U++ would compile with one of those new processors.

As of operating system, I am making use of the following:

Linux Mint 18.1 Cinnamon 64 bit
Cinnamon Version 3.2.7
Linux Kernel 4.4.0-53-generic

I ran GDB against the program as was able to retrieve the following information:

```
(gdb) run
Starting program:
/home/ptkacz/upp.out/reference/GCC.Debug.Debug_Full.Gui.Shared/GLDrawDemo
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
[New Thread 0x7fffea54c700 (LWP 5026)]
[New Thread 0x7fffe9d4b700 (LWP 5027)]
The program 'GLDrawDemo' received an X Window System error.
This probably reflects a bug in the program.
The error was 'BadValue (integer parameter out of range for operation)'.
  (Details: serial 3534 error_code 2 request_code 154 minor_code 3)
  (Note to programmers: normally, X errors are reported asynchronously;
   that is, you will receive the error a while after causing it.
   To debug your program, run it with the --sync command line
   option to change this behavior. You can then get a meaningful
   backtrace from your debugger if you break on the gdk_x_error() function.)
```

/usr/bin/xmessage

Thread 1 "GLDrawDemo" received signal SIGSEGV, Segmentation fault.
0x00000000005636bb in Upp::Ctrl::PanicMsgBox (title=0x7a91be "Fatal error", text=0x7b2b7f "Heap leaks detected!")
    at //home/ptkacz/upp/uppsrc/CtrlCore/GtkApp.cpp:56
56 __BREAK__;
(gdb) list
51   GtkWidget *dialog = gtk_message_dialog_new(NULL, GTK_DIALOG_MODAL, GTK_MESSAGE_ERROR,
52   GTK_BUTTONS_CLOSE, "%s: %s", title, text);
53   gtk_dialog_run(GTK_DIALOG (dialog));
54   gtk_widget_destroy(dialog);
55 }
56 __BREAK__;
57 }
58
59 void InitGtkApp(int argc, char **argv, const char **envptr)
60 {
(gdb)

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by Klugier on Fri, 30 Jun 2017 11:46:48 GMT
View Forum Message <> Reply to Message

Hello,

GTK backend supports OpenGL in very limited way - the main reason why it happens is lack of DHCtrl. The easiest fix for now is to re-switch to X11 backed (Just add this flag to your main package flags) and compile. Please notice that you will still need my patch to make this working.

I will check what is wrong there - at least it shouldn't crash The DHCtrl is hard to developer in GTK enviroment, but if we want to fully support OpenGL in this enviroment it is must have.

If you have any idea ant time to to fix our problems - the patch is welcome

Sincerely,
Klugier

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by ptkacz on Sun, 09 Jul 2017 19:31:27 GMT

View Forum Message <> Reply to Message

Where does one specify the DHCtrl flag?

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by Klugier on Sun, 09 Jul 2017 21:06:58 GMT
View Forum Message <> Reply to Message

Hello,

This is not DHCtrl flag - this is the back-end settings. Currently the default for POSIX like system is GTK. We need to change that using main package flag. You need to achieve the following result in the IDE:

Sincerely,
Klugier

File Attachments
1) Screenshot_20170709_230135.png, downloaded 940 times



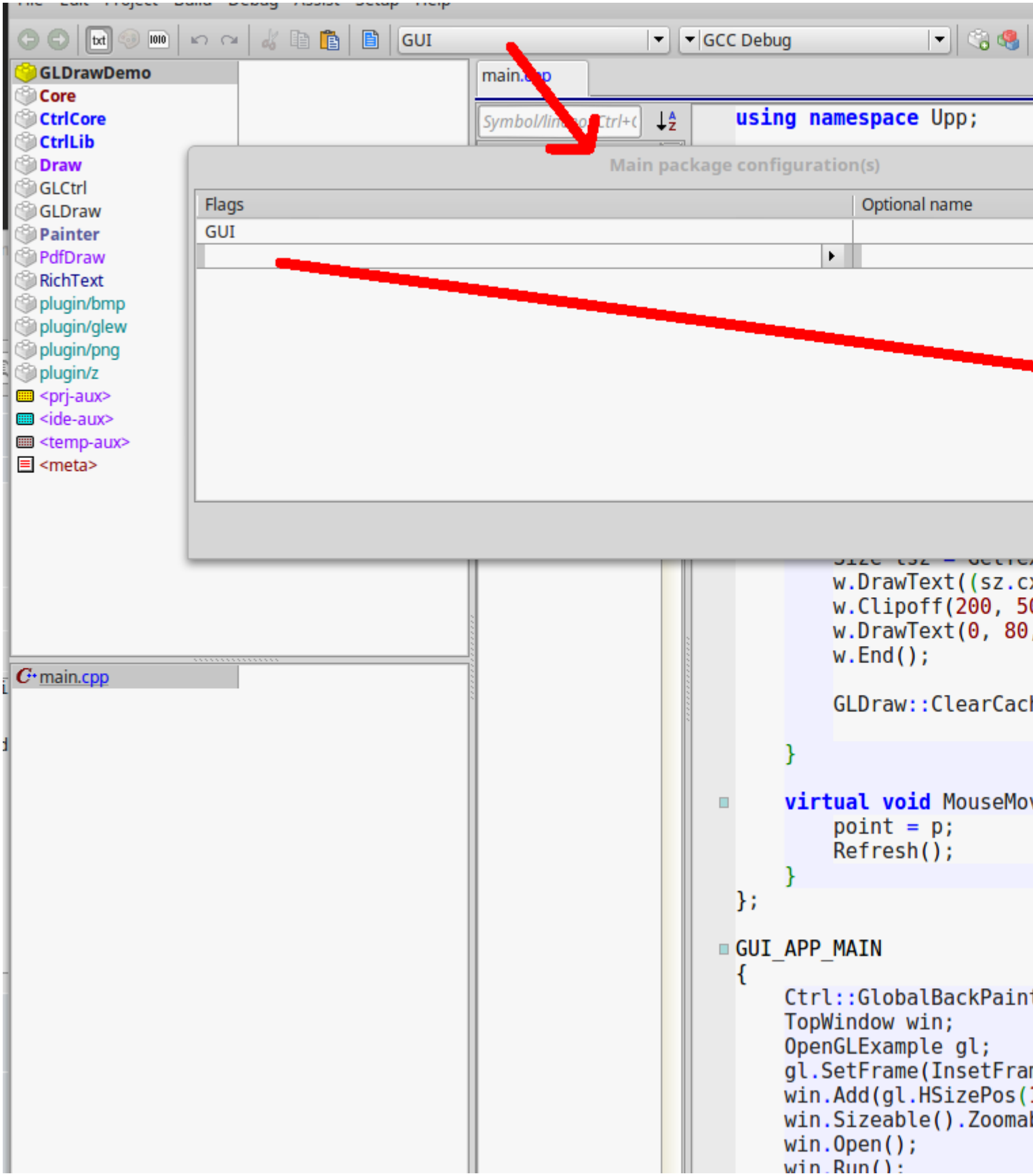
Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by ptkacz on Mon, 10 Jul 2017 03:16:25 GMT
View Forum Message <> Reply to Message

That seems pretty easy, but when I bring up the GLDrawDemo project, the only option that displays within the drop-down is, "GUI". Not other options list to select from.

When I click on the listed "GUI" option, a Main package configuration(s) dialog window lists with, GUI listing under the Flags heading. Appending a row the the list, and selecting the right selection triangle, a Configurations flags dialog window displays (see screen shot below).

How come the GUI X11 option didn't list for this project? How do I go about adding it, so that the project is correctly built?

File Attachments
1) ConfigurationFlags.png, downloaded 936 times


GUI
GCC Debug
GLDrawDemo
Core
CtrlCore
CtrlLib
Draw
GLCtrl
GLDraw
Painter
PdfDraw
RichText
plugin/bmp
plugin/glew
plugin/png
plugin/z
<prj-aux>
<ide-aux>
<temp-aux>
<meta>
main.cpp
using namespace Upp;
Main package configuration(s)
Flags
Optional name
GUI
main.cpp
w.DrawText((sz.cx
w.Clipoff(200, 50
w.DrawText(0, 80,
w.End();
GLDraw::ClearCach
}
virtual void MouseMov
point = p;
Refresh();
};
GUI_APP_MAIN
{
Ctrl::GlobalBackPaint
TopWindow win;
OpenGLExample gl;
gl.SetFrame(InsetFra
win.Add(gl.HSizePos(
win.Sizeable().Zooma
win.Open();
win.Run();

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by Oblivion on Mon, 10 Jul 2017 05:55:24 GMT
View Forum Message <> Reply to Message

Hello ptkacz,

For GUI, select GUI from the "App style" options.
For X11, simply select the NOGTK option.
For any other -custom- flag you'd like to activate, you can type it into "Other flags" section,
For example, you can explicity specify X11 flag (instead of using NOGTK option) by typing X11 into "Other flags" section too.

Best regards,
Oblivion

Subject: Re: OpenGL demo resulting in Fatal error, Heap leaks detected! error...
Posted by ptkacz on Tue, 11 Jul 2017 02:47:00 GMT
View Forum Message <> Reply to Message

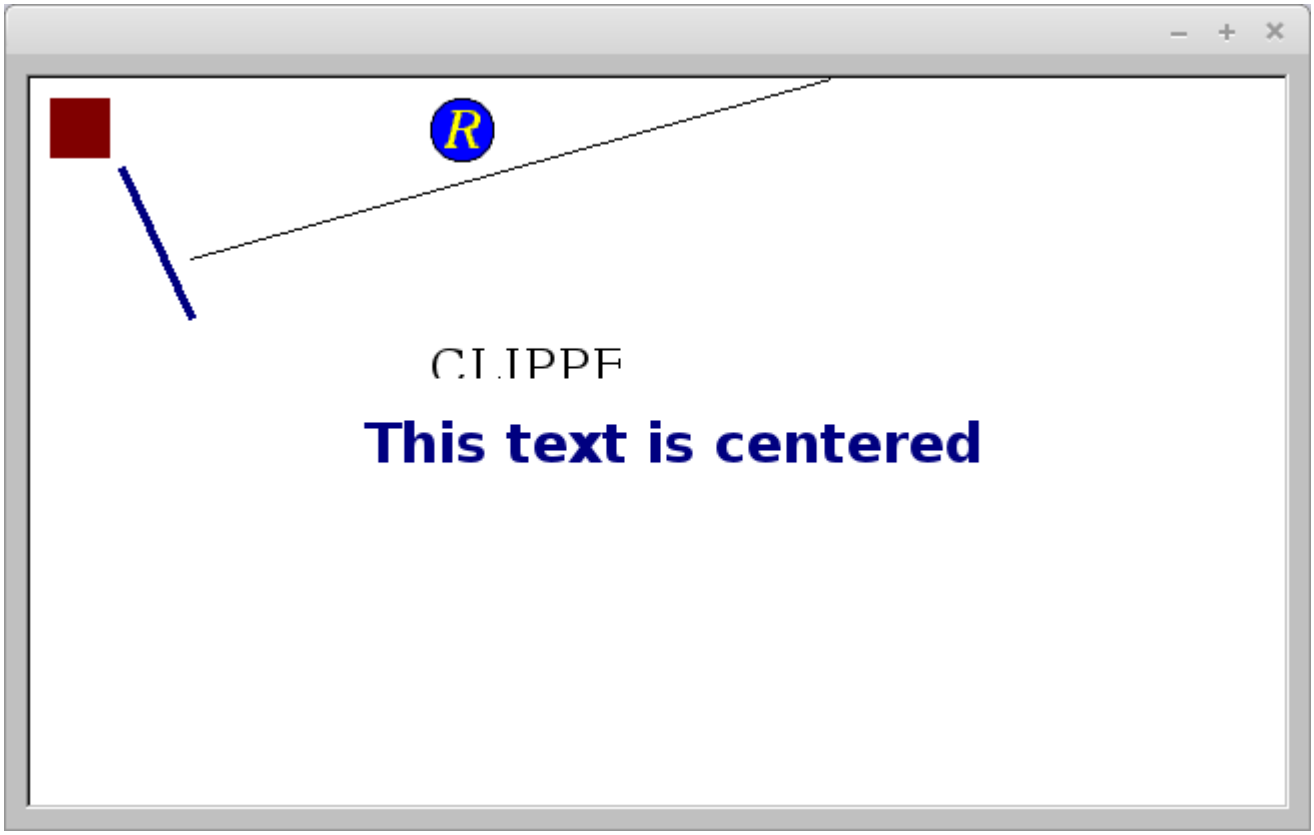
Thanks Guys!

That worked, selecting App Style to GUI and checking the NOGTK check box! Now displayed selected in the place of GUI within the drop-down is, GUI.NOGTK. Here's what resulted when the program compiled and ran:

I just went and tried the FTGL_Demo and that worked too!

Peter

File Attachments
1) GLDrawDemoScreenShot.png, downloaded 926 times


CLIPPE
This text is centered