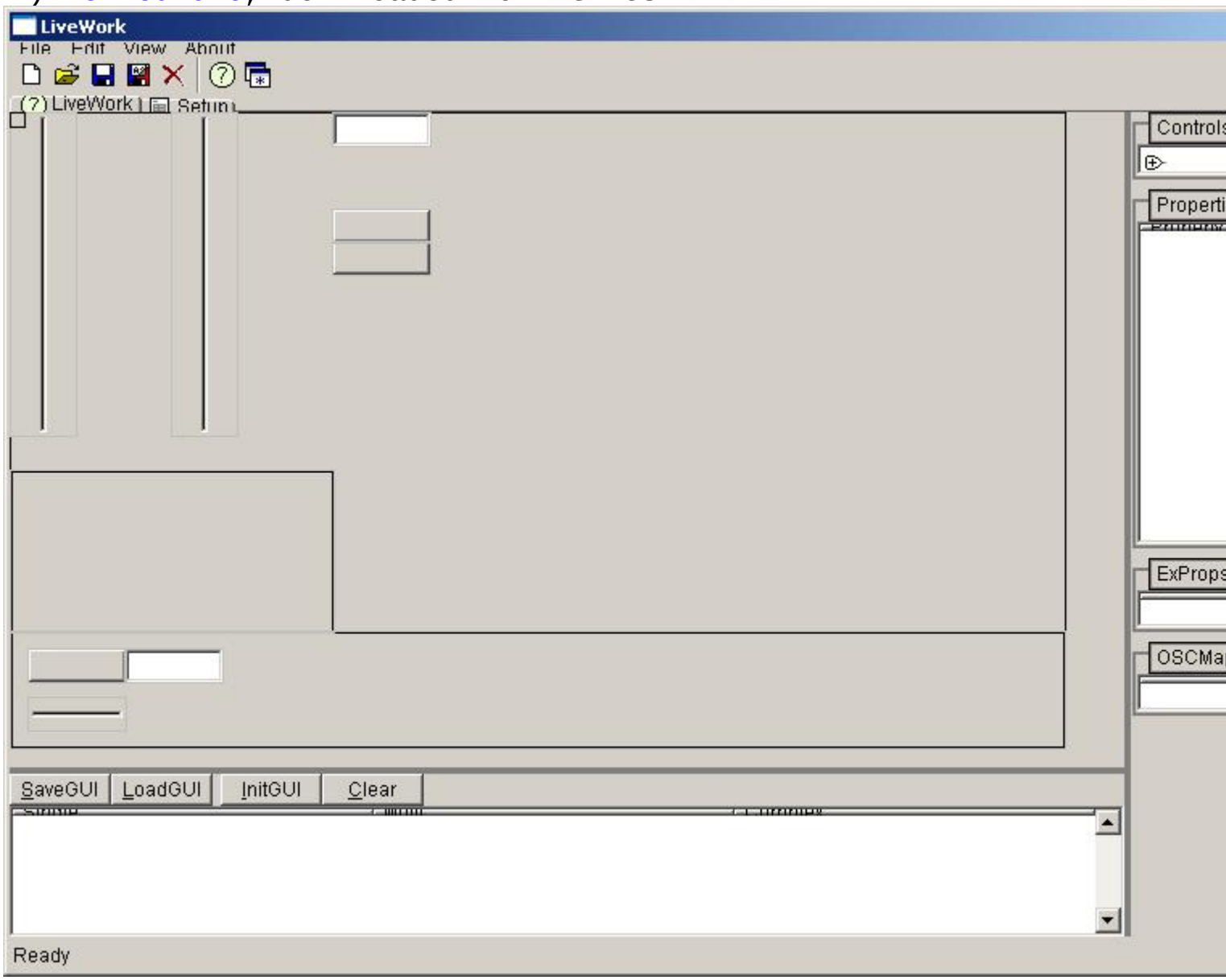
Subject: Weired appearance of GUI elements
Posted by kohait00 on Mon, 28 Mar 2011 06:00:12 GMT
View Forum Message <> Reply to Message

hi there,

im having this problem, and can't an explanation for it.. any ideas what's wrong? the sizes of the head spacers seem to be misleading..
it changes sometimes when adding new stuff to application..

## File Attachments

1) weired.JPG, downloaded 1047 times



Subject: Re: Weired appearance of GUI elements
Posted by MatthiasG on Mon, 28 Mar 2011 10:58:49 GMT
View Forum Message <> Reply to Message

hello,

i have seen something similar with the TabCtrls in my program. when compiled in debug mode the tab-headers are way too small. if i compile in optimal mode everything is fine though... couldn't find the source of this error till now though...

Subject: Re: Weired appearance of GUI elements
Posted by jerson on Mon, 28 Mar 2011 11:28:49 GMT
View Forum Message <> Reply to Message

I have seen this kind of behaviour when choosing auto-springs for widgets on the layout. If I use the left-top springs, nothing bad happens.

Subject: Re: Weired appearance of GUI elements
Posted by kohait00 on Mon, 28 Mar 2011 11:56:33 GMT
View Forum Message <> Reply to Message

well it pretty much looks like there is an issue with some kind of initial values then, or, even worse, memory corruption/overwrites..

Subject: Re: Weired appearance of GUI elements
Posted by dolik.rce on Mon, 28 Mar 2011 13:18:59 GMT
View Forum Message <> Reply to Message

Hi kohait

It doesn't look like a problem with springs, there are artifacts which are unrelated to springs (e.g. the height of menu bar)...

Is it reproducible? Did you try to compile on different machine or different options? Did you try full recompile (bomb button)? Could you post the code? (Or even better just a simplified example ).

Honza

Subject: Re: Weired appearance of GUI elements

Posted by MatthiasG on Mon, 28 Mar 2011 14:18:35 GMT
View Forum Message <> Reply to Message

Hi again,

i dont know, if my problem is the same as kohaits, but...
in tabctrl.cpp at the beginning there is something like:

```
CH_STYLE(TabCtrl, Style, StyleDefault)
{
 font = StdFont();
 tabheight = font.Info().GetHeight() + 8;
```

if i run my program in debug mode, the font.Info().GetHeight() call returns 0. Running in Optimal mode returns 13.
Couldn't reproduce this problem in a simple testcase though, so i think the error is somewhere in my own code...

Subject: Re: Weired appearance of GUI elements
Posted by kohait00 on Mon, 28 Mar 2011 16:08:03 GMT
View Forum Message <> Reply to Message

same with me here, in OPTIMAL using MSC this thing just looks as expected, debug spews the problems..
EDIT: and one more: TDM GCC yields the same thing in debug mode. so it has to be on code side somewhere, depending on debug.

Subject: Re: Weired appearance of GUI elements
Posted by MatthiasG on Tue, 29 Mar 2011 06:25:07 GMT
View Forum Message <> Reply to Message

ok, seems i found my error. had a static GridCtrl member in a class like in this thread:
http://www.ultimatepp.org/forum/index.php?t=msg&&th= 4820&goto=24145#msg_24145
Replaced it, and everything is ok now.
Perhaps your problem is the same kohait...

Matthias

Subject: Re: Weired appearance of GUI elements
Posted by kohait00 on Tue, 29 Mar 2011 09:25:24 GMT

thanks matthias,

i've posted a comment on this issue in the other thread. there is a simplified testcase which produces same results.

intrestingly enough, it doesnt occure always..depends on static init order..the old problem
thanks for the hint. i'll try to solve it somehow..

Subject: Re: Weired appearance of GUI elements
Posted by nlneilson on Fri, 17 Jun 2011 20:07:31 GMT


I had a similar issue as this.
In dlg.lay it looked OK.
When the app was run the EditField was about 6 times as high as it should have been, 3 up/down, that was white but the text was where it should be.

I just spent a bunch of time deleting that field, replacing, changing location, etc.. It just happened when it was in the third row down.

That was about 3 or 4 months ago, maybe the U++ update fixed it as there has been no problems since then.