Subject: global t_ macro screws up boost and my code...
Posted by kitovyj on Sun, 29 Nov 2015 16:57:58 GMT
View Forum Message <> Reply to Message

Hello guys,

I'm trying to use Upp as a GUI frontend for my app. But the globally defined t_ macro breaks boost bind header (it defines "t_" variable in one of the headers!). I tried to do "#undef t_" - but it didnt help(why?!). Probably BLITZ includes some Upp headers forcefully in each header? Have anybody tackled suchblem? Please help.

Best Regards,
Alexander

error messages:

```
In file included from C:\upp\uppsrc/Core/i18n.h:17:0,
                 from C:\upp\uppsrc/Core/Core.h:302,
                 from C:\upp\uppsrc/Draw/Draw.h:6,
                 from C:\upp\uppsrc/RichText/RichText.h:4,
                 from C:\upp\uppsrc/CtrlCore/CtrlCore.h:4,
                 from C:\upp\uppsrc/CtrlLib/CtrlLib.h:4,
                 from C:\projects\next\next.h:4,
                 from C:\projects\next\main.cpp:1:
c:\projects\boost_1_59_0/boost/core/ref.hpp: In constructor
'boost::reference_wrapper<T>::reference_wrapper(T&)':
C:\upp\uppsrc/Core/t_.h:9:24: error: class 'boost::reference_wrapper<T>' does not have any field
named 't_GetLngString'
 #define t_(x)         t_GetLngString(x)
                       ^
c:\projects\boost_1_59_0/boost/core/ref.hpp:73:57: note: in expansion of macro 't_'
    BOOST_FORCEINLINE explicit reference_wrapper(T& t): t_(boost::addressof(t)) {}
                                                        ^
```

Subject: Re: global t_ macro screws up boost and my code...
Posted by Didier on Sun, 29 Nov 2015 22:05:06 GMT
View Forum Message <> Reply to Message

Hello kitovyj,

I use boost and also ran into these kinds of problems.

The solution I found is to add the following line in the cpp file that includes and uses boost headers

#pragma BLITZ_PROHIBIT

==> the result is that this file IS NOT compiled with other arbitrary files by blitz so you don't get these errors

Hope it helps

Subject: Re: global t_ macro screws up boost and my code...
Posted by kitovyj on Mon, 30 Nov 2015 00:10:05 GMT
View Forum Message <> Reply to Message

It fixed the issue! Thank you, Didier.

Didier wrote on Sun, 29 November 2015 23:05Hello kitovyj,

I use boost and also ran into these kinds of problems.
The solution I found is to add the following line in the cpp file that includes and uses boost headers

#pragma BLITZ_PROHIBIT

==> the result is that this file IS NOT compiled with other arbitrary files by blitz so you don't get these errors

Hope it helps
