
Subject: TopWindow when Close override show exception unduly

Posted by [BetoValle](#) on Thu, 04 Feb 2021 14:51:43 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

when I create a simple application as shown below and rewrite the close method, when executing the application

memory exception is reported

(under windows 10 / 64bits /memory 8GB / TheIDE 15040 / memory free 42%)

```
#include <CtrlLib/CtrlLib.h>
```

```
using namespace Upp;
```

```
class SMain : public TopWindow {
```

```
public:
```

```
    typedef SMain CLASSNAME;
```

```
    SMain();
```

```
    void Close() override {
```

```
        delete this; // error show heap is corrupted !!!!
```

```
    }
```

```
};
```

```
SMain::SMain()
```

```
{
```

```
}
```

```
GUI_APP_MAIN
```

```
{
```

```
    SMain se;
```

```
    se.Run();
```

```
}
```

in the log file

Heap is corrupted --memory-breakpoint__ 3197158753

Memory at 0x000000000171FBF0, size 0x1 = 1

+0 0x000000000171FBF0 08

***** PANIC: Heap is corrupted --memory-breakpoint__ 3197158753

If I don't rewrite the method to close, there is no exception. This means a bug!

Important: I also noticed that if you add a layout containing a menu also with an option to close the screen pointed to the same method, the exception will no longer occur. Perhaps this is why this issue has not yet become evident.

thanks

Subject: Re: TopWindow when Close override show exception unduly
Posted by [Oblivion](#) on Thu, 04 Feb 2021 15:21:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello BetoValle,

```
void Close() override {  
    delete this; // error show heap is corrupted !!!!  
}
```

I don't think there is any bug in U++. You are trying to delete an object within the same object. That call to TopWindow::Close() will return to its caller, which is the TopWindow itself, and it will be a freed memory address. So you get heap corruption. Never do that. :)

Not to mention the SMain instance (se) is allocated on stack. You don't need to delete it anyway.

TopWindow::Close() method is useful for cleaning up your code, if required. It is not where you delete a window. In fact, unless it is absolutely necessary, we avoid using 'delete' in U++.

Best regards,
Oblivion

Subject: Re: TopWindow when Close override show exception unduly
Posted by [BetoValle](#) on Fri, 05 Feb 2021 02:48:13 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi, Thanks

i create this solution , after understand about ...

```

void Close() override {
    delete this;    // now it works because it was instantiated as a pointer!!!
}

GUI_APP_MAIN
{
    SMain* se;
    se = (new SMain);
    se->OpenMain();
    se->se.Run(); //Run when associated a especific window or "Ctrl::EventLoop()" when not associated
a window
        // and while any window open

}

```

Subject: Re: TopWindow when Close override show exception unduly
 Posted by [BetoValle](#) on Fri, 05 Feb 2021 12:22:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,
 (now I understood with you) not to "normally" rewrite the method. There is a better "as an" event
 to do something before closing the window:
 "WhenClosed" works even better, since I can incorporate some verification routine into it, and if I
 want to close the window, then I must call Close ().

```

WhenClose = [=] { doSomething; };
...
void SMain::doSomething()
{
    ...
    ...

    if(okGetout){ //and if necessary close
        Close();
    }
}

```

Subject: Re: TopWindow when Close override show exception unduly

Posted by [mirek](#) on Sun, 07 Feb 2021 09:31:21 GMT

[View Forum Message](#) <> [Reply to Message](#)

Oblivion wrote on Thu, 04 February 2021 16:21Hello BetoValle,

```
void Close() override {  
    delete this; // error show heap is corrupted !!!!  
}
```

I don't think there is any bug in U++. You are trying to delete an object within the same object. That call to TopWindow::Close() will return to its caller, which is the TopWindow itself, and it will be a freed memory address. So you get heap corruption. Never do that. :)

That is actually OK if it is the last thing you do to the object, regardless it being in the object's method. Both by C++ standard and U++ Close. Well, should probably be documented.. Similar approach is used here: [https://www.ultimatepp.org/examples\\$UWord\\$en-us.html](https://www.ultimatepp.org/examples$UWord$en-us.html)

In fact, "delete this" is sort of the only "delete" that is allowed in U++ :)

Quote:

Not to mention the SMain instance (se) is allocated on stack. You don't need to delete it anyway.

That is the real problem.

Mirek

Subject: Re: TopWindow when Close override show exception unduly

Posted by [Oblivion](#) on Sun, 07 Feb 2021 10:20:17 GMT

[View Forum Message](#) <> [Reply to Message](#)

Quote:That is actually OK if it is the last thing you do to the object, regardless it being in the object's method. Both by C++ standard and U++ Close. Well, should probably be documented.. Got that, and indeed it should be documented.

Best regards,
Oblivion
