
Subject: ToolBar Callback weirdness

Posted by [Alboni](#) on Fri, 27 Jan 2012 23:48:36 GMT

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

I created the next example to show off a problem I'm having with my app: (project added in zipfile)

For some reason the variable 'tool' is always 1 in the function ToolBarContent (callback for filling the toolbar) while it has the correct values everywhere else.

Is there some multithreading issue going on? I tried making 'tool' volatile but that didn't help.

Changing to multithreading library didn't change it either.

(I use Visual C++ 2010 Express)

-----example-----

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

```
using namespace Upp;
```

```
#define IMAGEFILE <bartest/bartest.iml>
```

```
#define IMAGECLASS bartest
```

```
#include <Draw/iml.h>
```

```
struct BarTest: TopWindow {
```

```
    ToolBar toolbar;
```

```
    int tool;
```

```
typedef BarTest CLASSNAME;
```

```
BarTest()
```

```
{
```

```
    tool=1;
```

```
    AddFrame(toolbar.Left());
```

```
    toolbar.Set(THISBACK(ToolBarContent));
```

```
}
```

```
virtual void Paint(Draw& w)
```

```
{
```

```
    char s[80];
```

```
    w.DrawRect(GetSize(), Gray());
```

```
    sprintf(s, "tool=%d", tool); // yeah I know
```

```
    w.DrawText(200, 220, s);
```

```
}
```

```
void MoveClicked()
```

```
{
```

```
    tool=1;
```

```
    Refresh();
```

```
}
```

```
void RotateClicked()
{
    tool=2;
    Refresh();
}

void ToolBarContent(Bar& bar)
{
    bar.Add("Move", bartest::move(), THISBACK(MoveClicked)).Check(tool==1);
    bar.Add("Rotate", bartest::rotate(), THISBACK(RotateClicked)).Check(tool==2);
}
};

GUI_APP_MAIN
{
    BarTest().Run();
}
```

File Attachments

1) [bartest.zip](#), downloaded 321 times
