
Subject: [Solved] Re: Is the Layout Designer outdated?
Posted by [Oblivion](#) on Tue, 18 May 2021 22:25:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello MeerMusik,

Quote:

Also what confused me for a while: In the C++ Header File I have to write "LabelBase Label1" but in the Layout File it is just Upp::Label - guess that is Part of the Charm of UPP.

Layout file is just a simple C++ file, there is nothing magical, no special processing, really. Only a couple of C++ macros (to simplify templating, and naming).

This:

```
struct MyWindow : TopWindow {
    Label label1;
    MyWindow()
    {
        Title("Window title").Sizeable().Zoomable().SetRect(0,0,640, 400);
        Add(label1. SetLabel(t_("I am a label")).SetInk(Color(226, 42, 0)).LeftPosZ(13, 243).TopPosZ(20, 144));
    }
};
```

Layout:

```
LAYOUT(MyWindowLayout, 640, 400)
    ITEM(Upp::Label, Label1, SetLabel(t_("I am a label")).SetInk(Color(226, 42, 0)).LeftPosZ(13, 243).TopPosZ(20, 144))
END_LAYOUT
```

Cpp:

```
#define LAYOUTFILE <MyWindow/MyWindow.lay>
#include <CtrlCore/lay.h>
```

```
class MyWindow : public WithMyWindowLayout<TopWindow> {  
public:  
    MyWindow()  
    {  
        CtrlLayout(*this, "Window title");  
    }  
};
```

The layout code does the ctrl instantiating for you. That's all. While the latter may seem complex at first glance, it actually simplifies things a lot, if you build complex GUIs. Not to mention it relies on C++ templating rules so the errors will be caught by the c++ compiler on compile-time.

Best regards,
Oblivion
