

---

Subject: Re: Widget events - any other than WhenAction ?

Posted by [peterh](#) on Sun, 04 Sep 2022 19:28:05 GMT

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

---

There is another - more elegant - solution:

Code your derived Ctrl as a template:

(this is experimental stuff I made for learning purposes, it works as expected, but does nothing meaningful)

```
template <typename TCtrl>
struct MyCtrl : public TCtrl {
    int count = 0;
    MyCtrl()//:TCtrl() // Default constructor wird automatisch aufgerufen, deshalb ist ...:MyCtrl:TCtrl()
(hier) nicht notwendig, schadet aber auch nicht.
    {
        WhenEnter << []{Cout() << " Enter "};
    }

    virtual void Paint(Draw& w) override {
        //w.DrawRect(GetSize(), White());
        TCtrl::Paint(w);
        w.DrawText(2, 2, AsString(count));
    }
};
```

Then you can derive arbitrary Controls from it, and the compiler will bark if there is a problem.

Then define your control in the layout editor like this:

I think this should be in the tutorial.

It might be "cold coffee" to regular users, but will get new users started.

An \*.usc file is not required in this case.

In most cases I want to expand or modify the behavior of an existing control.

I think this is a very common problem that new users will encounter and possibly will give up on it.

Then this comes in handy.

---

## File Attachments

1) [Layout2.jpg](#), downloaded 593 times

Search (Ctrl+F)

Gui21Layout

Type	Var / lbl
MyCtrl<EditString>	myctrl
EditString	
MyCtrl2	myctrl2

MyCtrl<EditString>    myctrl

SetFont    StdFont()

MaxChars   

AlignRight   

SetEditable   

SetFrame    default

Tip        Ctx    Id

WantFocus   

NotNull   

MaxLen   

TrimLeft   

TrimRight