

Hi,
you can use WidgetFactory callback:

.h

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

struct test : WithXXX<TopWindow> {
    typedef test CLASSNAME;
    test();
    void WidgetFactory(int line, One<Ctrl>& x);
    void OnButton(int n);
};
```

.cpp

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

using namespace Upp;

#include "test.h"

void test::OnButton(int n)
{
    PromptOK(AsString(n));
}

void test::WidgetFactory(int line, One<Ctrl>& x)
{
    Button& b = x.Create<Button>();
    b.SetLabel("Line " + AsString(line));
    b.LeftPos(10, 60).TopPos(4, 16);

    b <=< THISBACK1(OnButton, line);
}

test::test()
{
    CtrlLayout(*this, "Window title");
```

```
myArray.AddColumn("List of Tables").Ctrls(THISBACK(WidgetFactory));
```

```
myArray.SetLineCy( 20);
```

```
myArray.Add();  
myArray.Add();  
myArray.Add();  
myArray.Add();  
myArray.Add();  
}
```

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

Regards,
Omari
