Subject: Re: "Visual Inheritance" possible?
Posted by aschoem on Fri, 08 Jun 2007 15:27:32 GMT
View Forum Message <> Reply to Message

I'm a beginner with UPP so am not familiar with the template layout mechanism. Maybe you can explain the big advantage of this approach to me?

Maybe my description was a bit abstract so let me give an example. Again, I'm not familiar with UPP so have just taken some snippets from other examples to get something working, see below.

DialogBase shall be a reusable dialog component. Imagine it could have several controls and lots of methods. It could be an abstract base class as well.

The derived class Dialog_1 may have several additional controls and methods.

The points here are:
1) when designing the layout of derived dialogs I want to avoid code replication.
2) I might want to use polymorphism in derived dialogs as well.
3) Of course I can simply type this example with a text editor. However, the layout of the controls and their attributes, especially position and size, which may change for each derived dialog, can't be done easily with a text editor. That is where I need the layout designer.

So the question is: can the layout designer produce code that lets me specify the graphical layout in a simple way without compromising the code structure I want?

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

using namespace UPP;

class DialogBase : public TopWindow
{
 typedef DialogBase CLASSNAME;

protected:
 Button b_;
 void b_cb() { PromptOK("Button clicked."); }

public:
 DialogBase()
 {
  Title("DialogBase");
  SetRect(0, 0, 100, 100);
  b_.LeftPos(10, 30).TopPos(10, 30);
  b_.SetLabel("?");
  b_ <<= THISBACK(b_cb);
  Add(b_);
 }
```

```
};

class Dialog_1 : public DialogBase
{
protected:
 DocEdit e_;

public:
 Dialog_1()
 {
  Title("Dialog_1").Sizeable();
  SetRect(0, 0, 200, 200);
  b_.LeftPos(50, 100).TopPos(160, 30);
  b_.SetLabel("Dialog_1");
  e_.LeftPos(10, 180).TopPos(10, 140);
  Add(e_);
 }
};

GUI_APP_MAIN
{
 Dialog_1().Run();
}
```