Subject: Resizing a GUI object?
Posted by ptkacz on Sun, 10 Sep 2017 05:31:03 GMT
View Forum Message <> Reply to Message

Hi Guys,

When we create a U++ APP, a layout is created. For example, if our APP is named SimpleAPP, the layout SimpleAPPLayout gets created.

Within the SimpleApp class header file, the class gets defined as,
class SimpleAPP : public WithSimpleAPPLayout<TopWindow> {...
We might also define some private member variables:

```
DrawCanvas canvas;
WithButtonLayout<ParentCtrl> buttonPanel;
```

Within main.cpp, the mail GUI body looks as follows:

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

Within the SimpleAPP constructor, our APP is defined as follows:

```
SimpleAPP::SimpleAPP()
{
 CtrlLayout(*this, "Simple APP");
 Sizeable();
 canvas.SetRect(100, 0, 600, 500 );

 CtrlLayout(buttonPanel);
 buttonPanel.someButton << THISBACK(doSomething);

 Add(buttonPanel);
 Add(canvas);
 BackPaint();
}
```

When the APP's window gets resized or maxed, the Paint method of our APP's class gets called (i.e. void Paint(Draw& w)). I can retrieve a layouts size, etc. for example:
Size sz2 = buttonPanel.GetSize();
This can be done for other components (i.e. canvas). The width of the buttonPanel can be accessed as, sz2.cx.

How can one set the size of a widget since a SetSize method does not appear to exist? In this case, it would be liked for the buttonPanel to remain the same size, but have the canvas be

resized to fill the remaining space.

Peter
