
Subject: Re: How to get an "OnParentOpen" in a control

Posted by [mirek](#) on Tue, 29 Dec 2009 16:42:40 GMT

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

koldo wrote on Sun, 27 December 2009 16:05I believe you are approaching it from the wrong side.

Maybe a little bit more actual description of the problem would allow me to provide a better advice.
Hello Mirek

In StaticImage there is an option that lets an image to be the window background.

For this it is necessary that:

- the image control has to be "SizePos-ted()", so it fills all the parent canvas
- the image control has to be the first child to avoid to cover any other control

This can be made by hand:

- doing a control.SizePos() in the parent control constructor
- putting this control as the first one in the layout designer

What I was looking for is that the same child image control does it all without the need of any code in the parent.

Thank you
Koldo

I suggest this clean solution:

```
TopWindow& TopWindow::Background(const PaintRect& prect);
```

Either you can do actually two StaticImage classes (StaticImage widget and WindowBackgroundImage as Display with PaintRect operator), perhaps with some common code or ancestor, or you can even mix both, only sometimes StaticImage will act as display (again, via operator PaintRect) and sometimes as widget.

In fact, it even could act as both Not that it would have any practical use, but technically it is simple.

Is not it nice:

```
StaticImage myimage;
```

```
....
```

```
TopWindow win;
```

```
win.Background(myimage);
```

?

Mirek
