Posted by mirek on Wed, 20 Sep 2006 17:20:01 GMT

View Forum Message <> Reply to Message

James Thomas wrote on Wed, 20 September 2006 11:12Here is a quick example:

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

class AWindow : public TopWindow
{
public:
 typedef AWindow CLASSNAME;
 StaticRect r;

 AWindow()
 {
  Add(r);
  r.Color(SBlack);
  r.SetPos(r.PosLeft(10, 100), r.PosTop(10, 100));

  SetPos(r.PosLeft(0, 200), r.PosTop(0, 100));
 }

 // Events
 virtual void RightDown(Point p, dword keyflags)
 {
  PromptOK("Right mouse button event");
 }
};

GUI_APP_MAIN
{
 AWindow w;

 w.Run();
}
```

When it's run a window opens with a black rectangle in it. If you click outside the rectangle you get a prompt, inside nothing happens.

This seems like pretty unhelpful behaviour and I can't see any reason for it in the StaticRect source. LabelBox and StaticText correctly pass the event up to the parent window.

Is this the way it's supposed to work?

Yes.

Anyway, you can easily acquire opposite behaviour - just call IgnoreMouse for it (which is what

LabelBox and StaticText (LabelBoc base) do.

Sure, there is question whether that should not be the default behaviour. I never got cought by this, so we left it as it is.