Subject: Re: Help needed to deal with Display
Posted by mrjt on Tue, 31 Mar 2009 10:54:28 GMT
View Forum Message <> Reply to Message

JoseB wrote on Mon, 30 March 2009 11:39Hello,
I am new to Upp. I am trying to do my first steps.
Reading the Alex thread concerning droplist colors I decided to try it using a different display but I lost the selection bar. Please try the program that is in attachment. It seems that the blue bar (selection bar) is drawn behind Display painting. how to fix this?

By the way, which is the best way to change switch control fore color. I tryied with Display but it seems that switch controls do not support it.

Is this a bug?
How to set the colors of a DropList (this is actually not obvious, it took me some time to figure out):

```
struct DropListColorDisplay : Display
{
 Color fore;
 Color back;
 Font font;
 dword style;

 void Paint(Draw& w, const Rect& r, const Value& q, Color ink, Color paper, dword s) const
 {
  bool hasstyle = s & style;
  StdDisplay().Paint(w, r, q, hasstyle ? ink : fore, hasstyle ? paper : back, s);
 }

 DropListColorDisplay(Color _fore, Color _back, dword s = 0, Font _font = StdFont()) :
  fore(_fore), back(_back), style(s), font(_font) {}
};

void SetColor(DropList &list)
{
 Color fore(255,255,0);
 Color bk(0,255,0);
 static DropListColorDisplay display1(fore, bk, 0);
 static DropListColorDisplay display2(fore, bk, Display::CURSOR);
 list.ValueDisplay(display1);
 list.SetDisplay(display2);
}
```

Currently it is impossible to set the color of a Switch ctrl. However, I actually have had a small need for this for a while but hadn't got around to it so I will submit a patch to fix it. See attached files.

The patch adds a virtual PaintCase function that can be overloaded to give more control over how

the text is rendered. This isn't the usual Upp approach, but the Switch ctrl is kind of a special case because it's a cross between a data and Button-like ctrl.

After the path is applied you will be able to use the following class:

```
class ColorSwitch : public Switch
{
protected:
 Vector<Color> color;
 virtual void PaintCase(Draw &w, int i, int x, int y, int cx, const Font &font, bool disabled)
 {
  const Case &v = GetCases()[i];
  Color c = (i < color.GetCount()) ? color[i] : SColorLabel;
  DrawSmartText(w, x, y, cx, v.label, font,
                disabled ? SColorDisabled : c, ///////
                VisibleAccessKeys() ? v.accesskey : 0);
 }
public:
 void SetCaseColor(int i, Color c)  { ASSERT(i < GetCases().GetCount()); color.At(i, SColorLabel)
= c; }
 void Reset()      { color.Clear(); Switch::Reset(); }
};
```

Mirek: Could this patch be applied?

File Attachments

1) PushCtrl.h, downloaded 409 times
2) Switch.cpp, downloaded 622 times