
Subject: Re: Offset and the clipping stack. How to use?

Posted by [mirek](#) on Wed, 13 Jun 2007 11:09:10 GMT

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

mrjt wrote on Wed, 13 June 2007 06:32 Okay, I see how I can use GetClloffCount to record the stack depth and then trim it to the same level when I'm done, but I'm not sure why all this stack manipulation should be necessary for something so simple.

The main reason is to deliberately enforce that any painting (e.g. Paint) returns the offset and clipping to the state it started with.

You know, e.g., all widgets in a U++ window are painted using the single same Draw (there is just single WM_PAINT/Expose for all widgets). If any Paint routine would move offset out and not restored it, the painting result would be a mess...

Surely, there is an alternative solution that would use helper classes to represent "offseted" or "clipped" Draw, hard to say what is really better. In any case, stack in Draw is good enough to do the task.
