
Subject: Re: How would I virtualize a scrollable view to dynamically loadCtrls?

Posted by [mirek](#) on Tue, 31 May 2022 09:13:10 GMT

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

Placing those text in Paint is not a good idea. The time is spent in RichTextCtrl::GetHeight, which is slow. I have tried to optimise

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

using namespace Upp;

struct App : TopWindow {
    ScrollBar sb;
    Array<RichTextCtrl> items;

    int GetLineHeight() {
        return 25;
    }

    void Scroll() {
        Size sz = GetSize();
        int sy = sb;
        int y = 0;
        for(RichTextCtrl& item : items) {
            int h = item.GetRect().GetHeight();
            if(sz.cx && y + h > sy && y < sy + sz.cy) {
                item.Show();
                item.SetRect(0, y - sy, sz.cx, h);
            }
            else
                item.Hide();
            y += h + 5;
        }
    }

    void Paint(Draw& w) override {
        Size sz = GetSize();
        w.DrawRect(sz, SWhite());
    }

    void Layout() override {
        Size sz = GetSize();
        sb.SetPage(sz.cy);

        if(sz.cx) {
            int y = 0;
            for(RichTextCtrl& item : items) { // compute the height - that can be slow
```

```

Rect r = item.GetRect();
if(r.GetWidth() != sz.cx) {
    int h = item.GetHeight(sz.cx);
    item.SetRect(0, 0, sz.cx, h);
    item.Hide();
    y += h + 5;
}
}
sb.SetTotal(y);
}

Scroll();
}

void MouseWheel(Point, int zdelta, dword) override {
    sb.Wheel(zdelta);
}

bool Key(dword key, int) override {
    return sb.VertKey(key);
}

App() {
    Sizeable().Zoomable();

    for(int i = 0; i < 10000; i++) {
        RichTextCtrl& item = items.Create<RichTextCtrl>();

        item.SetData(
            Format(
                "[@(%d.%d.%d) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
tempor incididunt ut labore et "
                "dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi
ut aliquip "
                "ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu "
                "fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia
deserunt "
                "mollit anim id est laborum.]",
                (int)Random(255), (int)Random(255), (int)Random(255)
            );

        item.SetZoom(Zoom(1, 1));
        item.SetRect(0, 0, 0, 0);
        item.IgnoreMouse();

        Add(item);
    }
}

```

```
sb.Enable();
sb.WhenScroll = [this] { Scroll(); };
sb.SetLine(GetLineHeight());
AddFrame(sb);
}
};
```

```
GUI_APP_MAIN {
  App app;
  app.SetRect(0, 0, 250, 500);
  app.Run();
}
```

Which works fine for 10000 texts; with 100000 texts it is slow again, but the time once again is lost in GetHeight - that one basically needs to typeset that paragraph with all typographic rules. I can try to look into it, OTOH as we will need to support advanced composition in future, chances are it will only get slower then.

Mirek
