
Subject: Re: Alignment of two (or more) toolbars at the same "level"

Posted by [andrei_natanael](#) on Wed, 01 Sep 2010 12:54:34 GMT

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

281264 wrote on Tue, 31 August 2010 22:37Andrei,

Very grateful. It will be very useful.

What about the drag and drop feature?

Cheers,

Javier

Hello Javier,

If you want drag and drop you may use Docking from Bazaar, which provide rich docking options. If you want something simple, you may do it by your self

Here is my solution. Add an instance of BarHandler to a Bar (MenuBar, Toolbar) and see how it's working.

```
class BarHandler: public Ctrl {
```

```
public:
```

```
    Ctrl *parent;
```

```
    BarCtrl* bar;
```

```
    bool left_down;
```

```
    Point left_point;
```

```
    BarHandler() : bar(0), parent(0), left_down(false) {}
```

```
    void UpdateLayout()
```

```
    {
```

```
        if (bar) {
```

```
            int align = bar->GetAlign();
```

```
            if (align == BarCtrl::BAR_LEFT || align == BarCtrl::BAR_RIGHT)
```

```
                TopPos(1, 4).HSizePosZ(2, 2);
```

```
            else
```

```
                VSizePosZ(2, 2).LeftPos(1, 4);
```

```
        }
```

```
    }
```

```
    void ParentChange()
```

```
    {
```

```
        bar = 0;
```

```
        VSizePos(2, 2).LeftPos(1, 4);
```

```
    }
```

```

void InitCtrl()
{
    Ctrl* ctrl = GetParent();
    while (ctrl) {
        bar = dynamic_cast<BarCtrl*>(ctrl);
        if (bar && (bar->IsToolBar() || bar->IsMenuBar())) {
            parent = bar->GetParent();
            break;
        } else {
            bar = 0;
        }
        ctrl = ctrl->GetParent();
    }
}

void MouseLeave()
{
    if (left_down)
        SetCapture();
}

void LeftDown(Point p, dword keyflags)
{
    left_down = true;
    left_point = p;
    if (!bar || !parent)
        InitCtrl();
    if (bar && !bar->IsPopUp()) {
        Rect r = bar->GetScreenView();
        parent->RemoveFrame((CtrlFrame&)*bar);
        bar->SetRect(r.left, r.top, r.Width(), r.Height());
        bar->PopUp(NULL, true, true, true, true);
    }
}

void MouseMove(Point p, dword keyflags)
{
    if (bar && left_down) {
        Rect r = bar->GetScreenView();
        bar->SetRect(p.x - left_point.x + r.left, p.y - left_point.y + r.top, r.Width(), r.Height());
    }
}

void LeftUp(Point p, dword keyflags)
{
    left_down = false;
    int left, right, top, bottom;
    left = right = top = bottom = 0;
    if (bar && bar->IsPopUp()) {
        Rect pr = parent->GetRect();
    }
}

```

```

Point pt = GetMousePos();
if (pr.Contains(pt)) {
    bar->Close();
    parent->AddFrame(*bar);
    left = pt.x - pr.left;
    right = pr.right - pt.x;
    top = pt.y - pr.top;
    bottom = pr.bottom - pt.y;
    if (left < right && left < top && left < bottom)
        bar->Left();
    if (right < left && right < top && right < bottom)
        bar->Right();
    if (top < left && top < right && top < bottom)
        bar->Top();
    if (bottom < left && bottom < right && bottom < top)
        bar->Bottom();
}
}
ReleaseCapture();
UpdateLayout();
}

void Paint(Draw& w)
{
    ChPaint(w, GetSize(), CtrlImg::SmallDot());
}
};

```

Andrei
