
Subject: Re: MILESTONE: gtk3 replaces gtk2 as default linux backend

Posted by [Tom1](#) on Wed, 22 Apr 2020 12:08:14 GMT

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

Hi,

Here's a completing feature for RectTracker for tracking line drawing (TrackLine) and free-hand polyline (TrackRoute):

```
class LineTracker: public RectTracker{
public:
    int mode;
    Vector<Point> route;

    Vector<Point> TrackLine(const Point& p){
        mode=1;
        route.Clear();
        route.Add(p);
        Track(Rect(p,p),0,0);
        return clone(route);
    }

    Vector<Point> TrackRoute(const Point& p){
        mode=2;
        route.Clear();
        route.Add(p);
        Track(Rect(p,p),0,0);
        return clone(route);
    }

    void MouseMove(Point p, dword){
        if(mode==2 || route.GetCount()<2){
            route.Add(p);
        }
        else route[1]=p;
        Refresh();
        sync(Rect(route[0],route[route.GetCount()-1]));
    }

    void Paint(Draw& w){
        w.DrawImage(0, 0, master_image);
        w.Clip(clip & GetMaster().GetSize());
        int style=width;
        if(pattern!=DRAWDRAGRECT_SOLID){
            Color color2 = IsDark(color) ? White() : Black();
            w.DrawPolyline(route,width,color2);
            switch(pattern){
                case DRAWDRAGRECT_NORMAL:
```

```

        style=PEN_DOT;
        break;
    case DRAWDRAGRECT_DASHED:
        style=PEN_DASHDOT;
        break;
    }
}
w.DrawPolyline(route,style,color);
w.End();
}

LineTracker(Ctrl &master): RectTracker(master){
    MinSize(Size(-100000,-100000));
    mode=1;
}
};

```

Feel free to include it in U++.

To test the functionality, you can try:

```

void LeftDown(Point p, dword flags){
    LineTracker tracker(*this);
    tracker.Solid();
    tracker.Width(3);
    tracker.SetColor(Blue());

    Vector<Point> res;

    if(flags&K_CTRL) res = tracker.TrackRoute(p);
    else res = tracker.TrackLine(p);
    // ...
}

```

It proved easy enough to inherit from RectTracker, so I made it a separate class after all.

Best regards,

Tom
