Subject: Re: MILESTONE: gtk3 replaces gtk2 as default linux backend
Posted by Tom1 on Fri, 17 Apr 2020 11:20:27 GMT

View Forum Message <> Reply to Message

mirek wrote on Fri, 17 April 2020 01:41Tom1 wrote on Thu, 16 April 2020 21:08How would similar code and its performance look with what you refer to as 'Refresh and optimized Paint'?

So I guess we are in similar situation, painting maps and complex polygon. What we actually do is that we paint the map to Image, then in Paint we just put this image on screen and all "indicators" paint over that map.

Mirek

Well, so it seems. This basically sounds the same what I'm doing. My chain is vector map > Painter > ImageBuffer > SetSurface. And then the small moving targets or overlay texts/objects/indicators on top of that with a higher update rate. Just like my example above. The situation reminds me of sprites in the old times, which would fit in nicely. :)

The cost of 6-12 ms per 4K screen update (using SetSurface) for just a tiny little change is too much in my opinion. The new small-area ViewDraw based approach is far more efficient and should be here to stay even if MacOS does not support it at the moment -- or ever.

Another approach could possibly be to have 'Refresh(Rect) and Paint(Rect)/Paint(Vector<Rect>)' routine variants for partial updates via Paint. But I'm not sure if this solves the RectTracker requirements. And I would still prefer the new partial ViewDraw for the purpose...

Yet another question is, if OpenGL, Direct2D/3D, or some other backend would offer faster SetSurface capability taking the cost of 4K screen update well below millisecond level. Then again, I would like to avoid that path for now to ensure best compatibility with existing low-end hardware the clients may have.

Best regards,

Tom