
Subject: How to Draw into an existing image(buffer) ?
Posted by [nixnixnix](#) on Thu, 12 Jul 2007 14:37:23 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi everyone,

So, here's my problem:

- 1) I want to draw multiple vector and raster objects to a single view
- 2) I want to be able to control the opacity and/or blending method for each layer (perhaps not the vector layers although filled polygons yes)

I have written a function to draw an image to a Rect within another image and apply alpha blending, luminance, bitwise AND/OR etc. and it works fine although am sure someone here could optimise it to draw faster...

I already know that I need to use some sort of background image (have chosen an ImageBuffer) which then gets passed to a painting function for each layer which then does its stuff to the ImageBuffer before letting the next layer up have its go and so on.

My problem is that ImageDraw and DrawingDraw do not draw to already existing images. Its in the documentation and am sure its not impossible to implement as you can convert an Image to an ImageDraw object. However, in the Draw code there is only one-way conversion which means that any kind of complex drawing code is impossible unless I've missed something.

Surely ImageDraw needs to be able to draw to an image and not just create an image which is the case at present.

I can do my multiple raster layers as I have implemented my own function for drawing an image but it seems silly for me to have to re-implement the vector drawing functions as well.

I guess I could draw my vector layers to fresh images and then set an image mask and draw the new image to the main background ImageBuffer but that is a lot of processing for something that should be simple.

Would it be possible to have a DrawImage constructor or an = operator to go from Image or ImageBuffer to DrawImage please?

Cheers,

Nick

p.s. a search for ImageDraw on this site brings up only the documentation. I can't believe am the only person trying to do complex graphics on this platform!

