
Subject: Re: Chameleon again

Posted by [cbpporter](#) on Tue, 19 Jan 2010 07:16:02 GMT

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

andrei_natanael wrote on Mon, 18 January 2010 19:59cbpporter wrote on Mon, 18 January 2010 10:34

The two red spots are the hot spots. This divides the image into 9 areas. This image will be applied to all sizes of Button. The hot spots describe which areas will be resized. In this image, the cyan areas will remain constant size, the orange areas will be resized in one dimension, and the gray/purple one will be resized in both dimensions. Resizing algorithm is smooth and uses a filter which is appropriate for making good looking pictures while resizing. You will never get a blocky image, but sometimes you get a blurry one.

In that case with both hotspots having distinct locations and $x,y \neq 0$ seems logic how each part created by hotspot position is used to compose the button, but what about the following cases?

Here hotspots split image in 2, 3 or 4 parts not 9 as in your case.

I think when hotspots coincide in $x,y=0,0$ the image is scaled equally in both directions (it doesn't appear in image), but in other cases which part is used for which button part?

Well the simple answer is that you use hotspots to determine a rectangle that is going to be scaled and zones outside of that rectangle won't be scaled or scaled only in one dimension. It is useful because it allows you to have constant borders but smooth filling. You can play around around with the positions of hotspots and when they are identical you get the same result like if they were describing a symmetrical border. So the practical answer is: make a test application, use SetSkin with ChStdSkin or ChClassicSkin, edit the hotspots in CtrlLib/CtrlLib.iml and ClassicCtrls.iml, and after playing around for a while, you should get the feeling for hot spot functionality.
