Evas presentation, Part2: Graphic Primitives

David Odin

David@dindinx.org

Last month, we took a brief look at Evas' drawing capabilities. Today, we'll study in more details this library's graphic primitives and the six types of objects Evas supports.

Table of Contents

3. Rectangles

1. The shapes Evas uses

Evas’ canvas can display a number of objects of different types. Evas supports six types of objects:

.

4. Polygons

Evas can also display polygons without restrictions on the number of sides. A polygon is added to a canvas with the function:i[(f12.928.966 Tf 0 -21.619[(Evas)-2_Object)-600(e)-2_add_poly(Evas canvas);EvaThe polygon created will not have any ve polygon:i[(f12.928.966 Tf 0 -21.619[(Evavoid)-600(e)-2_add_point(Evas)-600(canvas,)]TJ 107.596 -11.656[(Evas)-2_Object object,E ’canvas’.EvaTo change the coordinates of the vertices of a polygon, the only solution is to remove all the points and re-add the vertices with the new coordinates with evas_add_point() function calls. The following function removes all the vertices of a polygon:i[(f12.928.966 Tf 0 -21.618[(Evavoid)-600(e)-2_clear_points(Evas)-601(canvas,)]TJ 123.735 -11.657[(Evas)-2_C

The fourth type of objects Evas can use is the color gradient. A gradient is really a rectangle filled with two-by-two color gradients in a given direction. Such a gradient is added to the canvas by the following function:i[(f12.928.966 Tf 0 -21.619[(Evas

To add an image object to a canvas, only the filename is needed:

```
Evas_Object evas_add_image_from_file(Evas canvas, char *filename);
```

Note that contrary to all Evas objects, image objects have by default the size of their image. Even if the file can’t be read, a completely transparent object with a 0x0 size is created anyway.

The evas_set_color() function doesn’t play the same role with image objects as with the other types of objects. For an image, it sets color and opacity values that will be used as a (multiplicative) filter for each pixel of the image. For example an evas_set_color(canvas, image, 0 ,0 ,255, 127); will make this image completely blue (since the red and green values will be multiplied by 0) and half transparent (127 begin about halfway between 0 and 255).

The file associated to an image object can be changed after its creation by this function:

```
void evas_set_image_file(Evas canvas, Evas_Object image, char * filename);
```

This is very useful to create animations.

By default, the image is scaled to completely fill the image object, even if the size of the object changes. To prevent this:

```
void evas_set_image_fill(Evas canvas, Evas_Object object,
                         double x, double y,
```

```
Evas_Object           line, rectangle;
Evas_Object           polygon, gradient;
Evas_Object           text, image;
Evas_Gradient         liste;
int                   i, j;

/* Creating the canvas */
canvas = evas_new ;
](the)-600(canvas)-60 ;
```