The contours-based gradients.

$$u(x,y) = \int U(v,w)e^{i(vx+wy)}\,dvdw = \int \frac{U(v,w)(v^2+w^2)}{(v^2+w^2)}dvdw; \qquad (1)$$

Here we start. This simple equation shows us the way how one can create contour-based gradients. Fourier transform of gradients of some function performs accordingly the next formulae:

$$\frac{\partial u(x,y)}{\partial x} = g_x(x,y) = \int iU(v,w)ve^{i(vx+wy)}dvdw = \int G_x(v,w)e^{i(vx+wy)}dvdw\,; \quad (2)$$

Combination of (1) and (2) gives us the next:

$$u(x,y) = \int \frac{(vG_x(v,w)+wG_y(v,w))e^{i(vx+wy)}}{(v^2+w^2)}dvdw; \qquad (3)$$

Therefore, if we have non-zero gradients, distributed along set of contours, corresponding colour (brightness, or something else) can be calculated according to (3). This complex transform can be replaced by two real ones, of course. Due to the fact, that large gradients for any photorealistic image are distributed along narrow lines, and their directions are at nearly perpendicular to these lines, there is no need to record both components of a gradient: only one scalar component, perpendicular to corresponding contour line. Hope, I've explained it enough clear. Assume, we have the only one Bezier curve- four control points. One scalar parameter is assigned to it, and this parameter changes from 0 to 1, when a point is moving along this curve from control point number 0 to control point number 3. Therefore, gradient can be described as a cubic spline, depending on before named scalar parameter. Well, how one can create a raster with such gradients? An empty complex matrix, initially filled by zeroes, is to be created. A Bezier curve is to be drawn at that canvas- if a curve goes through some cell of this matrix, gradient value is to be placed to this cell. Then complex forward Fourier transform is to be performed. Then resulting complex matrix is to be transformed again according to (3), what is actually a backward transform. Because kernel of this transform is a transform of fundamental solution of Laplasian equation, this method is based actually on this equation. A non-zero density, distributed along set of contours, is convolved with Kauchy's kernel.

The vertices-based gradients.

There is no way to use Laplasian equation here. Some wise boys claim, that they do it, but it obviously a mistake. In case of 2D event solution of Laplasian equation, if resulting function adopts some predefined values at set of separated from each to all others control points is a constant except theses control points. It is a pure math, nothing to do. For this event a biharmonic equation must be used. There is an analogue- let's try to imagine a flat table and set of nails, hammered into it. Each nail's head has its own elevation above table's surface. Let's take a flat thin flexible resistible piece of steel (tinfoil) and press it to theses nails. Each point of this tinfoil will have its own elevation above the table. This is what biharmonic equation gives us. Problem is to solve this equation (and solve it quickly). There are some more or less efficient methods. Relaxation method is one of the simplest, but after some enhancement could be rather efficient too. Decrease matrix in size, start with one eighth of the original size, make some iterations, use obtained result as the first approximation to the one fourth matrix and so on. Another possible approach, efficient if number of control points is relatively small- up to 400-500 is to reduce partial differential equation (what a biharmonic equation actually is) to system of linear equations. There are more methods, but I cannot tell now, which is the best. Some experiments must be done to find an answer to this question. This sort of gradient is simpler than the contour-based one, at least additional records what must be included into SVG specification are simpler. The only one record must be included, something like this:

<vertices V123 77 0.73 V244 879 0.33 … V4 49 -1.13 /vertices>

If this record is placed inside some path, gradient fills will be realized inside a region, bounded by this path, otherwise to the whole image. Result of such interpolation may be not only colour distribution, but any scalar value as well. This type of a record can be used in science application, for example in meteorology. If one needs to paint a region, corresponding record may be the next:

<gradient>

V123 77 #AAFF07 V244 879 #FF00FF … V4 49 00FF00 /vertices>

</gradient>

Conversion raster images to vector file with such gradients is the other story, but here I wantedonly to explain how such gradients could be rasterized.

Andrey Matseevskiy, Kluchi, Kamchatka.