

## Structuring Graphical Data

June 22, 1998

[Lisa Rein \(/pub/au/5\)](#)

Up to this point, XML has been used to identify the structure of text elements in Web documents. Now there is a growing demand to use XML to define the syntax for graphical information in Web documents as well.

### Vector and Raster Graphics

The vector approach is different from raster or bitmap images, such as GIF, which are widely used on the Web. Raster images literally define every pixel they contain. A bitmap image is defined in much the same way as images on the screen of a TV, pixel by pixel. BMP, TIFF and JPEG graphics are other examples of "bitmapped" formats.

The term "vector" comes from the days when plotters created graphics by establishing coordinate positions and then drawing lines between them or filling in the area that the points contain. Vector graphics are mathematical descriptions of graphical elements that enable a viewer (such as a browser) to display lines, curves, and text at optimal resolution, whether viewing or printing. For example, to "draw" a circle, you call the "circle" function and supply the desired dimensions as arguments.

Consider how these two different approaches might represent a painting. A vector graphic would specify a set of "instructions" or separate strokes required to draw the painting, specifying the direction, thickness and other properties of each stroke. A bitmapped format would specify the sequence of individual pixels (or other units) and their state -- whether the pixel is "on" or "off," or which should be red, green, or blue. Every last one of these pixel positions must be defined, so raster files tend to be larger than vector files. They also tend not to scale without loss of data that can affect the resolution and color of the image.

However, even bitmaps can be represented in XML. The example below shows how tedious it is, requiring the textual representation of every single pixel.

```
<pixel coords="x,01" pixel coords="y, 20" setting="line"/> <pixel coords="x,0  
  pixel coords="y, 20" setting="line"/> <pixel coords="x,03" pixel coord  
  setting="line"/> <pixel coords="x,04" pixel coords="y, 20" setting="li
```

An XML representation of a vector format can be a lot more interesting, and usually a lot more compact depending upon what's being drawn:

```
<line startcoord="x,01 y,01" endcoord="x, 20 y, 20" thickness="3"/>
```

By storing a visual image as a mathematical description of its lines, curves, shapes and colors, a vector graphics format is compact and efficient. More importantly, the same image can be rendered at optimal resolution on a computer screen or in print.

## Why Structure Graphical Data?

Representing graphical data in a structured manner has many potential benefits:

- Extracting text from graphics for indexing
- Building indices of graphical components.
- Distinguishing between informational graphics and navigational graphics.
- Dynamically generating or modifying graphical elements.
- Controlling the positioning and layout of graphics through style sheets.
- Creating interactive graphical elements with scripting.

An *XML-enabled* vector-based graphical language can include all kinds of information that might be delivered along with the graphic or be available depending upon what request was made. It can provide the ability to add "extra" information about graphical data for "free." Adding metadata to graphics can open the door to new applications without significantly adding to the document's file size or download time. A graphic might supply metadata such as author, title, date and copyright. It might supply the aspect ratio or physical size for a stand-alone print. It might supply business information such as fees for usage.

In summary, an XML-based vector graphics language would make Web graphics lightweight elements of a Web document, and bring with it greater functionality and flexibility.

---

Content licensed from and © 1998 - 2008 O'Reilly Media, Inc.