

#\$+K!

## Manual for wxSVGFileDC

by Chris Elliott

### Contents

Copyright notice  
wxSVGFileDC

---

C ontents  
C ontents  
b rowse00001  
K Contents  
D isableButton("Up")

**Copyright notice**

---

Copyright notice  
topic0  
browse00002  
Copyright notice  
DisableButton("Up")

\$#+K! **wxSVGFileDC**

wxSVGFileDC

---

w\_xSVGFileDC  
t\_opic1  
b\_rowse00003  
K\_wxSVGFileDC  
D\_isableButton("Up")



## \$#+K1. **wxSVGFileDC**

A `wxSVGFileDC` is a *device context* onto which graphics and text can be drawn, and the output produced as a vector file, in the SVG format (see <http://www.w3.org/TR/2001/REC-SVG-20010904/> ). This format can be read by a range of programs, including a Netscape plugin (Adobe), full details at <http://www.w3.org/Graphics/SVG/SVG-Implementations.htm>8 Vector formats may often be smaller than raster formats.

The intention behind `wxSVGFileDC` is that it can be used to produce a file corresponding to the screen display context, `wxSVGFileDC`, by passing the `wxSVGFileDC` as a parameter instead of a `wxSVGFileDC`. Thus the `wxSVGFileDC` is a write-only class.

As the `wxSVGFileDC` is a vector format, raster operations like `GetPixel` are unlikely to be supported. However, the SVG specification allows for PNG format raster files to be embedded in the SVG, and so bitmaps, icons and blit operations into the `wxSVGFileDC` are supported.

A more substantial SVG library (for reading and writing) is available at <http://www.xs4all.nl/~kholwerd/wxstuff/canvas/htmldocbook/aap.html>

### **Derived from**

---

`wxSVGFileDC`  
`wxSVGFileDC`  
`browser00004`  
`K wxSVGFileDC`  
`E nableButton("Up");ChangeButtonBinding("Up", "JumpId(^svg.hlp', `topic1')")`