SVG in Fritzing: A Case Study

by

Jonathan Cohen - jonathan@fritzing.org
Mariano Crowe - merunga@fritzing.org
Brendan Howell - brendan@fritzing.org
SVG in Fritzing: A Case Study

or...
SVG in Fritzing: A Case Study

or...

SVG and Fritzing: A Love/Hate Relationship
What's Fritzing?
What's Fritzing?
What's Fritzing?
Why SVG for Fritzing?

Bitmaps:
• Poor performance
• Aliasing artifacts when scaling
Why SVG for Fritzing?

SVG:
- Fast rendering
- Scaling with no degradation
- Real world units
- Our users already use SVG editing tools
- XML manipulation
<xml version='1.0' encoding='UTF-8'>
<module ... >
  <meta-data>
    <.../>
  </meta-data>
  <views>
    <iconView
      image="icon/LED-red-5mmicon.svg" .../>
    <breadboardView
      image="breadboard/LED-5mm-red.svg" .../>
    <schematicView
      image="schematic/led.svg" .../>
    <pcbView
      image="pcb/T1.75_LED.svg" .../>
  </views>
  <connectors>
    <.../>
  </connectors>
</module>
SVG/Sketch Relation: Layers

```xml
<svg >
  <g id="silkscreen">...
  </g>
  <g id="copper0">...
  </g>
</svg>
```
Parts Problems

- We can't make all the parts users will need
- Make it easy for the users to create new parts
Solution: Parts Editor

- Create art with familiar tools (inkscape & illustrator)
- Parts editor handles xml markup so users don't have to
Solution: Parts Editor

- Create art with familiar tools (inkscape & illustrator)
- Parts editor handles xml markup so users don't have to

DEMO
Solution: Parts Editor

- Create art with familiar tools (inkscape & illustrator)
- Parts editor handles xml markup so users don't have to
Complications:
• illustrator idiosyncrasies
• inkscape idiosyncrasies
• Qt idiosyncrasies
Parts Editor

Complications:
- illustrator idiosyncrasies
- inkscape idiosyncrasies
- Qt idiosyncrasies
Export

- svg
- png
- jpg
- ps
- pdf
- Gerber
Steps:

- Normalize elements
- Translation to absolute coordinates
- Convert wires to line elements
- Change *strokes and fill* attributes
- Hide unused layers
- Gerber “walkthrough”
Exporting to Gerber

Fritzing

Gerber Viewer
Love:
• Fast rendering
• Smooth scaling
• Real world units
• Programmatic manipulation of images

Hate:
• No standard standard (validation)
• No standard toolkit (c++)
THANKS!

www.fritzing.org
code.google.com/p/fritzing